ISSN 2224-087X

ЕЛЕКТРОНІКА ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

ELECTRONICS AND INFORMATION TECHNOLOGIES

Збірник наукових праць

Випуск 34

2026

ELECTRONICS AND INFORMATION TECHNOLOGIES

ЕЛЕКТРОНІКА ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Issue 34

Scientific journal

Published 4 issue per year

Published since 1966

Випуск 34

Збірник наукових праць

Виходить 4 рази на рік

Видається з 1966 р.

Ivan Franko National University of Lviv

Львівський національний університет імені Івана Франка

2026

ЗАСНОВНИК: ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА

Друкується за ухвалою Вченої Ради
Львівського національного університету
імені Івана Франка
протокол №8/6 від 23.06.2026 р.

Реєстрація суб'єкта у сфері друкованих медіа:
Рішення Національної ради України з питань
телебачення і радіомовлення № 1877 від
30.05.2024 р. Ідентифікатор медіа R30-04912

У 1966–2010 рр. збірник виходив під назвою «**Теоретична електротехніка»**

Збірник «Електроніка та інформаційні технології» містить оригінальні результати досліджень з електронного матеріалознавства, моделювання фізичних явищ, процесів і систем електроніки, обробки сигналів і зображень, інформаційних технологій.

“Electronics and information technologies” journal contains original research results on electronics material science, modelling of physical phenomena, processes and electronic systems, signal and image processing and information technologies.

Адреса редакційної колегії:
Львівський національний університет імені Івана Франка, факультет електроніки та інформаційних технологій, вул. Ген. М. Тарнавського, 107, 79017, Львів, Україна
тел. (+38) (093) 864-01-19

Editorial office address:
Ivan Franko National University of L'viv,
Faculty of Electronics and Computer Technologies
107 Tarnavsky St., UA–79017,
Lviv, Ukraine
tel. (+38) (093) 864-01-19

e-mail: elit@lnu.edu.ua
web-сайт: http://publications.lnu.edu.ua/collections/index.php/electronics/index

АДРЕСА РЕДАКЦІЇ, ВИДАВЦЯ І ВИГОТОВЛЮВАЧА:

Львівський національний університет імені Івана Франка
вул. Університетська, 1, 79000 Львів, Україна

Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру видавців, виготівників і розповсюджувачів видавничої продукції. Серія ДК № 3059 від 13.12.2007 р.

CONTENTS

ЗМІСТ

ОГЛЯДОВІ СТАТТІ

ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ

МОДЕЛЮВАННЯ ПРОЦЕСІВ ТА ЯВИЩ

МАТЕРІАЛИ ЕЛЕКТРОННОЇ ТЕХНІКИ

ISSN 2224-088X (print) • ISSN 2224-0888 (on-line)

Electronics and Information Technologies, 2026, 34, 5-30
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.932

IMAGE FUSION TECHNIQUES USING POLARIZATION INFORMATION AND INFRARED INTENSITY

Vitalii Artym**, **Oleh Krupych
Department of Optoelectronics and Information Technologies
Faculty of Electronics and Computer Technologies
Ivan Franko National University of Lviv,
107 Tarnavsky Str., Lviv, 79017, Ukraine

Artym, V.I., Krupych, O.M. Image Fusion Techniques Using Polarization Information and Infrared Intensity. *Electronics and Information Technologies*, 34, 5–30. https://doi.org/10.30970/eli.34.1

ABSTRACT

Background. Fusion of polarization images with conventional thermal infrared images combines complementary physical information to improve scene interpretation. Infrared intensity imaging captures thermal emission and emissivity patterns. In contrast, polarization parameters, including the degree and angle of linear polarization, encode surface microstructure, material composition, and geometric orientation that are inaccessible to intensity-only sensing. Their fusion enhances spatial detail, target discrimination, and robustness under degraded visibility conditions, reflecting a broader shift toward multimodal perception.

Materials and Methods. This study provides an analytical review of polarization–infrared fusion techniques. The development of methods is examined from classical linear transforms, multiresolution decomposition, and energy-based decision rules to adaptive hybrid frameworks. Particular attention is given to deep learning architectures and physics-informed models that incorporate polarization formation principles and radiometric constraints. The review situates fusion progress within advances in sensor technologies and computational imaging.

Results and Discussion. The analysis indicates that the incorporation of polarization information substantially enhances fused image quality compared with conventional intensity-based techniques. Classical algorithms provide stable and interpretable performance but exhibit limited adaptability to complex environments. Learning-based and hybrid strategies demonstrate superior contrast preservation, structural detail enhancement, and target detectability, especially in thermally heterogeneous or visually degraded scenarios. Persistent challenges include limited annotated datasets, polarization-induced noise sensitivity, and reduced physical interpretability of deep neural representations.

Conclusion. Polarization–infrared fusion represents a transition from mono-modality to multi-modal observation with the system-oriented perception integrating physical modeling with data-driven learning. Future research should emphasize physically reasoned neural design, standardized evaluation protocols, and improved interpretability to support reliable real-world deployment.

Keywords: image fusion, polarization imaging, infrared imaging, multimodal data, deep learning.

INTRODUCTION

Image fusion – understood as the synthesis of heterogeneous sensory information into a unified perceptual or computational representation – has become a foundational topic

in optical engineering and computer-vision research. Among the most powerful combinations is the fusion of infrared (IR) intensity and polarization imaging, two modalities that offer inherently coupled complementary physical descriptions of a scene. Infrared imaging provides thermal or emissive contrast independent of visible illumination, making it highly effective under low-light or adverse weather conditions [1]. Polarization imaging, by contrast, encodes the orientation and shape of the electromagnetic field vector oscillations, revealing surface texture, material properties, and scattering phenomena that are inaccessible to conventional intensity images [2].

Historically, fusion methods have evolved from simple arithmetic combinations toward sophisticated adaptive and learning-based schemes. In the 1990s and early 2000s, pyramid and wavelet transforms dominated, reflecting the influence of multiresolution analysis on computer vision. With the advent of polarimetric sensors and computational optics, the possibility of integrating polarization cues into traditional intensity images emerged. The initial goal was enhancement – improving contrast and edge visibility under adverse conditions such as fog, haze, or camouflage [3].

Subsequent decades witnessed the expansion of this field toward autonomous sensing, surveillance, and biomedical diagnostics. Yet the fusion of polarization and infrared modalities also signifies a conceptual evolution: it marks a transition from passive observation to active reconstruction of the scene's physical reality. Each modality expresses a distinct physical ontology – thermal emission versus electromagnetic wave polarization– whose synthesis mirrors the epistemological shift from single-perspective to multimodal cognition.

The following sections of this paper trace this evolution in both technical and conceptual terms. Section 1 outlines the formation of polarimetric and infrared imaging paradigms; Section 2 reviews classical, hybrid, and deep-learning fusion techniques; Section 3 examines applications and benchmarking practices; Section 4 discusses epistemological and methodological dimensions; and Section 5 concludes with perspectives on future integration.

1. POLARIMETRIC AND INFRARED IMAGING: PHYSICAL BACKGROUND

1.1. Historical Evolution of Polarization and Infrared Imaging

The convergence of polarization and infrared imaging is not merely a technological synthesis but the outcome of a long historical process in which distinct scientific traditions – wave optics, thermodynamics, radiometry, sensor engineering, and computational imaging – gradually intersected. A historiographical perspective reveals that polarization–infrared fusion emerged through successive epistemic phases: classical theoretical formalization, instrumental stabilization, digital transformation, and algorithmic integration. Each phase redefined both what could be measured and how measurements were interpreted.

Classical Optics and the Conceptual Birth of Polarization

The origins of polarization science lie in nineteenth-century wave optics. The decisive theoretical advance was the formulation of reflection and refraction laws for polarized light by Augustin-Jean Fresnel [4]. Fresnel's equations demonstrated that electromagnetic waves interacting with material boundaries exhibit polarization-dependent reflection coefficients. This discovery transformed polarization from a curious optical phenomenon into a quantitatively predictable property of wave – matter interaction.

The formal mathematical framework was followed with the introduction of the Stokes parameters by George Gabriel Stokes in 1852 [5]. The Stokes vector representation provided a complete description of partially polarized light through four measurable quantities (S_0, S_1, S_2, S_3). Importantly, this formulation unified intensity and polarization into

a single algebraic structure, enabling subsequent generations of researchers to treat polarization as a vectorial extension of radiometric measurement.

Throughout the late nineteenth and early twentieth centuries, polarization research developed largely within laboratory physics and astronomy [6]. Applications included mineralogy, crystallography, and atmospheric scattering studies. However, imaging was not yet central; measurements were typically pointwise and instrument-based, relying on rotating analyzers and photometric detection. The absence of array detectors constrained polarization analysis to experimental optics rather than scene-based imaging.

Historiographically, this period may be characterized as conceptual consolidation: the physical and mathematical principles of polarization were firmly established, but technological limitations prevented their integration into practical imaging systems.

Infrared Radiation: From Discovery to Radiometric Formalization

Infrared science followed a parallel yet distinct trajectory. The discovery of infrared radiation by William Herschel in 1800 [7] extended the electromagnetic spectrum beyond visible light. However, it was not until the development of thermodynamic theory that infrared radiation acquired quantitative meaning.

The formulation of blackbody radiation laws by Max Planck at the turn of the twentieth century [8] provided the theoretical basis for thermal radiometry. Planck's law established a direct relationship between temperature and spectral radiance, enabling the use of infrared measurements as thermometric indicators.

The early twentieth century saw incremental advances in infrared detectors, including bolometers and thermocouples [9]. These devices were sensitive but slow, limiting spatial resolution and dynamic imaging capability. Military applications during World War II accelerated infrared technology development, leading to scanning thermal imagers for target detection and night vision.

By the 1970s and 1980s, focal plane arrays (FPAs) based on materials such as HgCdTe and InSb enabled two-dimensional infrared imaging [10]. Long-wave infrared (LWIR) systems (8–14 μm) became especially prominent due to their compatibility with ambient-temperature thermal emission. The introduction of uncooled microbolometers in the 1990s democratized infrared imaging by reducing system size and cost [11].

From a historiographical standpoint, infrared imaging transitioned from laboratory thermometry to operational sensing through the stabilization of detector technologies. Unlike polarization science, which matured theoretically early but technologically late, infrared imaging experienced rapid technological advance driven by defense and industrial demand.

Digital Imaging and the Sensor Revolution

The late twentieth century introduced a transformative element common to both domains: digital imaging. The development of CCD and CMOS sensor arrays enabled spatially resolved acquisition of optical intensity [12]. For infrared systems, hybrid FPAs provided pixel-level radiometric sampling; for visible-spectrum systems, CMOS sensors allowed integration of micro-optical elements.

Polarization imaging underwent a comparable revolution with the invention of micro-polarizer arrays deposited directly onto sensor pixels [13]. These arrays consisted of sub-pixel polarization filters oriented at discrete angles (e.g., 0°, 45°, 90°, 135°), enabling snapshot acquisition of multiple polarization states without mechanical rotation.

This innovation resolved a long-standing instrumental limitation: polarization could now be measured spatially and temporally in dynamic scenes. Consequently, polarimetry transitioned from complicated laboratory photometry to real-time imaging.

The historical significance of this stage lies in the instrumental convergence of modalities. For the first time, polarization and infrared imaging shared comparable spatial resolution and digital representation, making computational integration feasible.

Early Multisensor Fusion: Transform-Based Integration

Before the specific integration of polarization and infrared modalities, multisensor fusion emerged as a broader research area in the 1980s and 1990s. Theoretical tools from multiresolution analysis played a decisive role.

The Laplacian pyramid method introduced by Peter J. Burt and Edward H. Adelson [14] provided a systematic framework for decomposing images into frequency bands. Shortly thereafter, the multiresolution wavelet formalism developed by Stéphane Mallat [15] established a mathematically rigorous basis for hierarchical image representation.

Fusion algorithms employed coefficient-selection rules such as maximum absolute value, local energy, or weighted averaging [16]. These approaches treated input modalities as statistically comparable signals, abstracting away their physical origins.

When polarization information was first incorporated into fusion pipelines, it was often reduced to scalar derivatives such as maps of *Degree of Linear Polarization* (DoLP), treated analogously to intensity images. The deeper physical implications of polarization – vectorial boundary interactions and anisotropic emission – were not yet fully integrated into algorithm design.

This period may be interpreted as algorithmic abstraction: fusion was understood primarily as signal combination in a transform domain, rather than as physically constrained integration.

Emergence of Polarimetric Thermal Imaging

The true convergence of polarization and infrared modalities required technological advances enabling simultaneous measurement of radiance and polarization state in thermal wavelengths. Polarimetric thermal imagers capable of estimating Stokes vector elements S_0, S_1, and S_2 began to appear in the early twenty-first century [17].

Research demonstrated that thermal emission from surfaces can exhibit partial linear polarization due to angular emissivity dependence and surface anisotropy. This insight revealed that polarization contrast could enhance target detection even when temperature contrast was minimal.

In camouflage scenarios, for example, thermally equilibrated objects might remain indistinguishable in intensity images yet display polarization signatures arising from reflective coatings or geometric discontinuities. Such findings stimulated the development of fusion algorithms specifically designed for polarization–infrared integration.

Unlike earlier multisensor fusion tasks, this integration required reconciliation of two distinct physical models:

- Radiometric emission is governed by Planck's law and emissivity functions.
- Vectorial polarization is governed by Fresnel boundary conditions.

Thus, fusion became a problem of heterogeneous physical synthesis, not merely signal combination.

Learning-Based Integration and the Data-Driven Turn

The 2010s introduced deep learning into image fusion research. Convolutional neural networks (CNNs) demonstrated the capacity to learn hierarchical feature representations directly from data [18]. Unsupervised fusion architectures eliminated the need for handcrafted coefficient rules, instead optimizing reconstruction losses and structural similarity metrics.

The publication of transformer-based architectures by Ashish Vaswani and colleagues [19] further advanced cross-modal feature integration through attention mechanisms. In polarization–infrared fusion, attention modules dynamically weight thermal and polarization features according to contextual saliency.

Recent developments incorporate physics-informed constraints into neural architectures [20]. For example, loss functions may enforce consistency with Stokes parameter relationships or penalize physically implausible polarization amplification. This

marks a shift toward hybrid epistemology, in which data-driven inference is constrained by physical law.

Standardization, Benchmarking, and Reproducibility

A crucial development in the maturation of polarization–infrared fusion research has been the emergence of standardized datasets and evaluation metrics. Early studies were limited by proprietary data and inconsistent benchmarking protocols.

Contemporary research increasingly employs multi-criteria evaluation frameworks incorporating entropy, mutual information (MI), structural similarity (SSIM), and visual information fidelity (VIF) [21]. More recently, physically motivated metrics assess whether fused outputs preserve thermal contrast and polarization integrity [22].

This standardization reflects the institutionalization of the field. Fusion research has moved from exploratory experimentation to systematic comparative evaluation, enabling reproducibility and cross-study validation.

Historiographical Evolution of Polarization–Infrared Fusion

From a historiographical perspective, the evolution of polarization–infrared fusion can be interpreted as a sequence of paradigm transitions:

- *Conceptual Phase* – Establishment of wave and radiometric theory (nineteenth–early twentieth century).
- *Instrumental Phase* – Stabilization of detector technologies and array imaging (mid–late twentieth century).
- *Digital Phase* – Emergence of computational imaging and multiresolution analysis (1990s).
- *Integrative Phase* – Physical and algorithmic convergence of polarization and infrared modalities (2000s).
- *Data-Driven Phase* – Deep learning and physics-informed models (2010s–present).

Each phase expanded not only technological capability but also epistemic scope. Polarization imaging transitioned from laboratory optics to dynamic scene analysis; infrared imaging evolved from thermometric measurement to semantic perception; fusion unified these modalities into integrated perceptual systems.

The convergence of polarization and infrared imaging thus exemplifies the broader transformation of optical science into computational multimodal sensing. What began as independent theoretical traditions has become a unified research domain characterized by interdisciplinary integration of physics, engineering, and machine learning.

1.2. Physical and Mathematical Principles

The polarization state of light can be fully described by the *Stokes vector* $\mathbf{S} = [S_0, S_1, S_2, S_3]^{\mathrm{T}}$, where S_0 denotes total intensity, S_1 and S_2 represent intensity differences of linear polarization components at orthogonal orientations ($S_1 = I(0°) - I(90°)$ and $S_2 = I(45°) - I(135°)$), while $S_3 = I_{Right} - I_{Left}$ corresponds to the intensity difference of right- and left-handed circular polarizations. For most passive imaging systems operating in the visible or infrared range, the Stokes parameter S_3 is irrelevant or unattainable; thus, the focus lies on the *Degree of Linear Polarization* (DoLP) and the *Angle of Polarization* (AoP) [23]:

$$\begin{aligned} DoLP &= \frac{\sqrt{S_1^2 + S_2^2}}{S_0}, \\ AoP &= \frac{1}{2}\tan^{-1}\left(\frac{S_2}{S_1}\right). \end{aligned} \tag{1}$$

Infrared detectors measure spectral radiance

$$L(\lambda, T) \approx \varepsilon(\lambda) \cdot B(\lambda, T), \tag{2}$$

where $\varepsilon(\lambda)$ is emissivity and $B(\lambda, T)$ is the Planck function.

The polarized component arises from anisotropic emission or reflection and carries information about surface orientation and roughness.

The challenge of fusion lies in reconciling these two data spaces – radiometric (temperature/emissivity) and vectorial (polarization). While IR intensity follows scalar thermodynamic laws, polarization obeys vector optics. Successful fusion therefore, demands both physical modeling and statistical optimization.

2. EVOLUTION OF IMAGE FUSION TECHNIQUES

The historical development of image-fusion techniques can be read as a gradual transition from *analytical* to *synthetic* paradigms of vision: from deterministic transforms to adaptive and data-driven inference. Each period reflects a different understanding of how information from distinct modalities – infrared intensity and polarization – should be interpreted and unified.

2.1. Classical Transform-Based Methods

Early methods were rooted in linear systems theory and multi-resolution signal analysis. Infrared and polarization images were decomposed into spatial–frequency representations using transforms such as the *Laplacian pyramid, discrete wavelet transform* (DWT), or *nonsubsampled contourlet transform* (NSCT). Fusion rules were then applied to corresponding coefficients.

Yue and Li [24] employed an oriented Laplacian pyramid to integrate infrared intensity with polarization information. The fusion rule selected coefficients of higher gradient energy, producing improved edge contrast and object detectability. Zhang et al. [25] later extended this approach using an orthogonality-difference metric that captured the divergence between polarization components, achieving sharper boundaries in long-wave infrared (LWIR) imagery.

Experimental studies on LWIR polarimetric imaging further illustrate the complementary nature of infrared intensity and polarization features. For example, the work mentioned above of Zhang et al. [25] demonstrates how polarization measurements at different analyzer orientations can be combined with infrared intensity to derive physically meaningful polarization descriptors such as the degree of linear polarization (DoLP) and the angle of polarization (AoP). These features highlight surface reflectance and material properties that do not show up in intensity images alone.

Figure 1 is a representative example of long-wave infrared polarization imaging and fusion [25]. Fragments (**Fig. 1 a–d**) show the raw measurements acquired by a polarization camera: the infrared intensity image and three polarization-filtered images captured at analyzer angles of 0°, 60°, and 120°. From these measurements, the polarization parameters DoLP and AoP are computed, as shown in (e) and (f). The final fused image (g) integrates thermal intensity and polarization information, revealing enhanced structural detail and improved discrimination of objects against the background. This example demonstrates how polarization descriptors can provide additional contrast cues that complement thermal information, motivating the development of classical fusion algorithms based on transform-domain representations.

Such approaches are interpretable and computationally efficient, yet they depend heavily on hand-crafted heuristics (e.g., choosing maximum or weighted average coefficients). They also assume stationary statistics across the image, which rarely holds for natural or cluttered scenes. A comparative summary of the main classical transform-

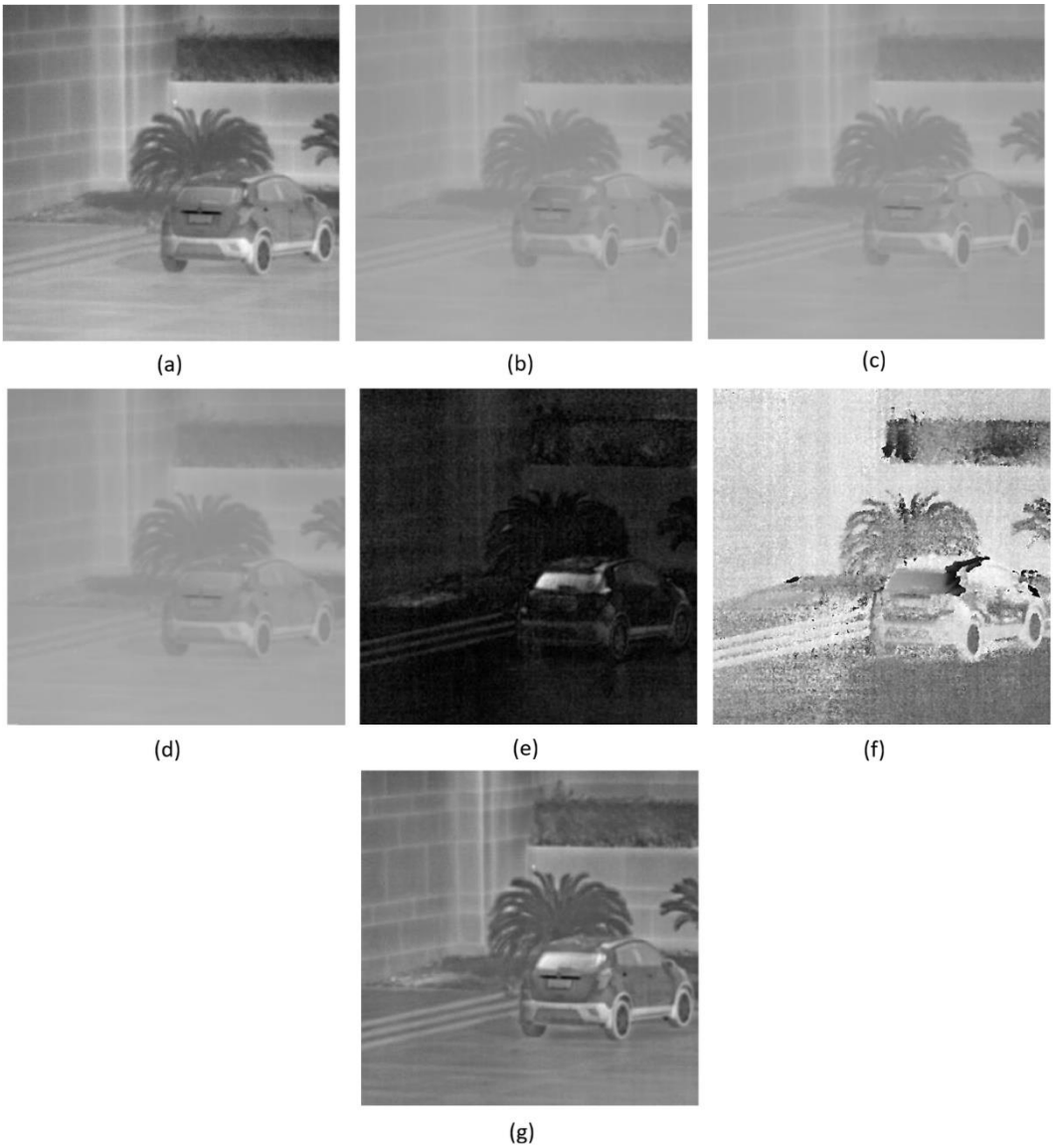

Fig. 1. Example of long-wave infrared polarization image fusion [25]: (a) infrared intensity image, (b) 0° polarization image, (c) 60° polarization image, (d) 120° polarization image, (e) degree of linear polarization (DoLP), (f) angle of polarization (AoP), and (g) fused image combining infrared intensity and polarization information

based fusion methods, including their representative transformations, fusion rules, strengths, and limitations, are provided in **Table 1**.

Laplacian Pyramid

Principle.

The Laplacian pyramid is one of the earliest multiresolution image representations, originally introduced by Burt and Adelson [14]. The method constructs a sequence of progressively smoothed (Gaussian-blurred) images and then computes the difference between consecutive levels, producing a set of band-pass (Laplacian) components. This representation is localized, efficient, and offers a multi-scale decomposition of the image.

Table 1. **Comparison of Classical Fusion Approaches**

Method	Transform Domain	Fusion Rule	Strengths	Limitations
Laplacian Pyramid [1]	Multiscale Spatial	Max Gradient	Simple, fast	Sensitive to noise
Wavelet Fusion [3]	DWT	Weighted Energy	Good localization	Artifacts at edges
NSCT [4]	Contourlet	Regional Entropy	Multi-directional	Computationally heavy

Algorithm (simplified):

1. Construct the Gaussian pyramid:
   $G_0 = I$,
   $G_{k+1} = \mathcal{D}(G_k)$ – smoothing + downsampling.
2. Compute the Laplacian levels:
   $L_k = G_k - \mathcal{U}(G_{k+1})$,
   where $\mathcal{U}$ is upsampling with interpolation.
3. Store all Laplacian levels $L_0, \dots, L_{N-1}$ and the coarsest Gaussian level G_N. Reconstruction is performed by iteratively adding back the upsampled higher-level images.

Fusion rules:

- Maximum absolute value per coefficient.
- Weighted average based on local variance or gradient energy.
- Region-based selection.

Advantages:

- Simple, interpretable, low computational cost.
- Preserves edges and local contrast effectively.

Limitations:

- Sensitive to noise (especially at high-frequency levels).
- No directional representation; cannot capture oriented structures well.

Discrete Wavelet Transform (DWT)

Principle.

The discrete wavelet transform provides an orthogonal or biorthogonal multiresolution decomposition, splitting the image into one low-frequency subband (LL) and three high-frequency directional subbands (LH, HL, HH). The theoretical foundation was introduced by Mallat through the concept of multiresolution approximation [15].

Algorithm:

1. Perform an L-level DWT on each input image to obtain LL, LH, HL, and HH subbands.
2. Fuse the low-frequency LL subbands using weighted averaging, entropy-based measures, or PCA-style rules.
3. Fuse the high-frequency subbands using:
   – Max-absolute-value selection,
   – Energy-based or gradient-based measures,
   – Local saliency criteria.
4. Apply inverse DWT to reconstruct the fused image.

Advantages:

- Strong spatial–frequency localization.
- Widely used in remote sensing, medical imaging, and multisensor fusion.

- Flexible choice of wavelet bases.

Limitations:

- Limited orientation diversity (only three high-frequency directional bands).
- Shift-variant – even small misalignments cause artifacts.
- Prone to aliasing when the sampling geometry changes.

Nonsubsampled Contourlet Transform (NSCT)

Principle.

The nonsubsampled contourlet transform (NSCT), proposed by da Cunha, Zhou, and Do [26], is a shift-invariant, overcomplete, directional multiresolution transform specifically designed to represent edges and contours more effectively than wavelets. Unlike the classical contourlet transform, NSCT avoids downsampling, making it fully translation-invariant.

NSCT consists of two key components:

1. Nonsubsampled Pyramid (NSP): Multiscale decomposition without subsampling.
2. Nonsubsampled Directional Filter Bank (NSDFB): Directional decomposition at each scale, also without subsampling.

Fusion rules:

- High-frequency directional subbands: saliency- or energy-based selection.
- Low-frequency subband: weighted or statistical combination.

Advantages:

- Excellent directional and anisotropic representation of curved structures.
- Shift-invariant (no downsampling), reducing misregistration artifacts.
- Highly effective for preserving edges and textures.

Limitations:

- High computational cost and memory usage due to redundancy.
- Quality depends strongly on filter design.
- Requires parameter tuning (number of scales/orientations).

2.2 Multi-Parameter and Hybrid Polarization Fusion

The next phase introduced multi-parameter models, integrating DoLP, AoP, and sometimes partial Stokes information with IR intensity [17]. The guiding idea was that different polarization parameters represent complementary structural cues.

An important direction within hybrid and physically interpretable fusion methods involves the use of complex-valued representations and matrix formalisms for combining visible, infrared, and polarization data. Khaustov and co-authors proposed several approaches based on complex functions, Jones-matrix formalism, and complex-vector fusion models, demonstrating improved target saliency preservation and adaptive fusion behavior in military sighting systems [27–33].

Unlike single-parameter fusion approaches that rely solely on DoLP, multi-parameter fusion exploits the complementary physical information encoded in different polarization descriptors. Specifically, DoLP highlights material-dependent reflection properties, AoP conveys geometric and orientation-related surface characteristics, while infrared intensity provides robust thermal contrast under low-visibility conditions. Hybrid fusion frameworks typically combine these parameters at different representation levels, including pixel-level weighted fusion, multiresolution transform domains (e.g., wavelet- or contourlet-based fusion), and feature-level strategies guided by saliency or edge measures. More recently, deep learning–based methods have been introduced to adaptively learn fusion rules across multiple polarization parameters and infrared intensity, resulting in improved target discrimination and structural preservation. Nevertheless, multi-parameter polarization fusion remains challenged by polarization noise, parameter normalization, and the physical interpretability of learned fusion mechanisms, motivating ongoing research into physics-informed and noise-aware fusion models.

2.3 Emergence of Learning-Based Paradigms

With the rapid development of machine learning – particularly convolutional neural networks (CNNs) and attention-based architectures – image fusion has entered an adaptive, data-driven era. Unlike classical fusion methods that rely on hand-crafted rules or fixed transform-domain strategies, learning-based approaches automatically learn nonlinear mappings between multimodal inputs and fused outputs through end-to-end optimization.

Unsupervised and Self-Learned CNN Fusion. Li et al. proposed an unsupervised CNN-based fusion framework, known as DenseFuse, in which an encoder–fusion–decoder architecture learns effective fusion representations without requiring ground-truth fused images [18]. The method employs structural similarity–based loss functions to preserve salient features from multiple modalities while maintaining visual consistency. This approach demonstrated that meaningful fusion rules can be learned directly from data, marking a key transition from handcrafted fusion strategies to representation-learning paradigms.

Attention-Driven and Transformer-Based Models. Recent studies have incorporated attention mechanisms to enhance the selective integration of multimodal features. Attention-guided networks dynamically emphasize salient regions – such as thermally prominent targets – while retaining fine structural details derived from polarization cues [34]. These mechanisms improve robustness in complex scenes by adaptively weighting feature contributions across spatial locations and modalities.

A representative example is the attention-guided filtering network proposed by Li et al. [34] for infrared–polarization image fusion. **Figure 2** presents a qualitative comparison of fusion results obtained using several representative algorithms. The attention-guided approach demonstrates improved preservation of structural details and enhanced contrast of thermal targets while suppressing noise in polarization measurements. This illustrates the advantage of attention mechanisms in selectively integrating complementary information from infrared intensity and polarization features.

Physics-Aware Deep Fusion. Beyond purely data-driven learning, recent efforts have explored the integration of physical priorities into deep fusion frameworks. Noise-aware and physics-consistent loss formulations, particularly for polarization–infrared fusion, have been shown to improve stability and interpretability while preserving the advantages of deep learning [34, 35]. Such approaches constrain network behavior in accordance with polarization imaging physics and reduce artifacts caused by noisy polarization measurements.

Overall, learning-based fusion methods demonstrate superior adaptability and robustness compared to classical techniques. However, their performance is strongly dependent on data availability, which remains limited for polarization – infrared imaging. Consequently, current research increasingly focuses on unsupervised, semi-supervised, and physics-informed learning strategies.

3. COMPARATIVE ANALYSIS OF APPROACHES

3.1 Qualitative and Quantitative Evaluation

Evaluating the quality of fusion between polarization and infrared images is a nontrivial task, since for such multimodal systems, there is no universal ground-truth image that simultaneously and fully represents all physical properties of the observed scene. Unlike reconstruction or segmentation problems, where an objectively correct result exists, image fusion quality is determined by the degree of preservation, integration, and coherence of information originating from multiple sources. Consequently, fusion evaluation criteria are inherently multidimensional and encompass informational, structural, physical, and perceptual aspects [36].

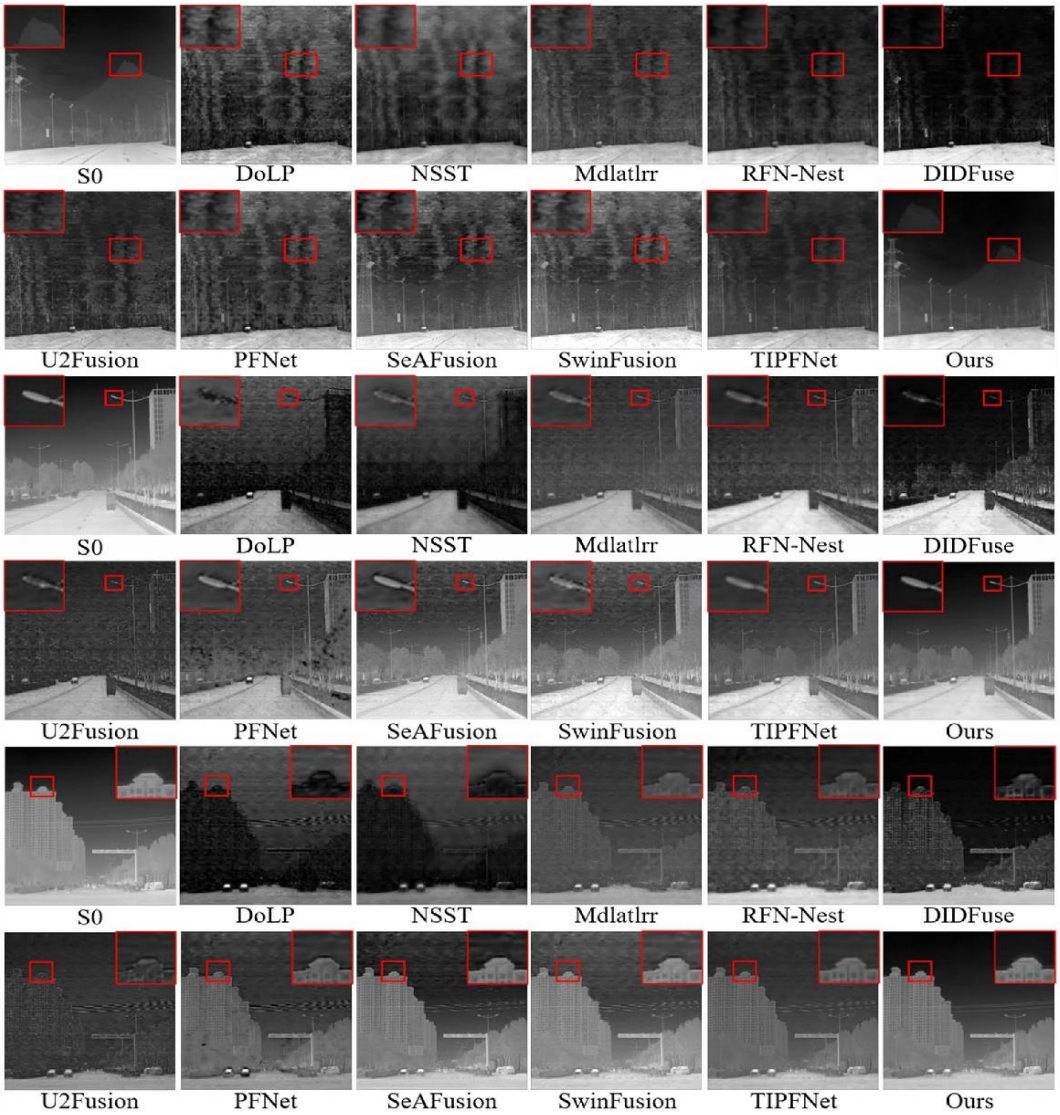

Fig. 2. Visual comparison of polarization–infrared image fusion results produced by different algorithms [34]

Informational Criteria

Informational metrics assess how effectively the fused image preserves the informational content of the source modalities. One of the most commonly used measures is entropy, which characterizes the statistical diversity of intensity levels in the image. In the context of polarization–infrared fusion, increased entropy typically indicates the incorporation of additional textural and structural details, particularly those derived from DoLP or AoP parameters. However, excessive entropy growth may also reflect noise amplification, which is especially pronounced in polarization channels [36].

A more informative measure is mutual information (MI) between the fused image and each input modality. High MI values indicate effective transfer of informative components from both the infrared intensity and polarization parameters. Contemporary studies often employ normalized or summed MI formulations, enabling consistent comparison across different scenes and fusion algorithms [37].

Modern deep-learning-based fusion studies also rely on information-theoretic criteria to validate performance improvements over classical approaches [38].

Structural and Spatial Criteria

Structural metrics are aimed at assessing the preservation of spatial patterns, contours, and local scene features. The average gradient (AG) is a simple yet effective indicator of edge sharpness and fine-detail representation. For polarization–infrared fusion, a high AG value generally correlates with accurate reproduction of geometric structures, which are often emphasized in AoP maps or polarization contrast images.

A more advanced metric is the structural similarity index (SSIM), which evaluates local similarity in terms of luminance, contrast, and structural components [39]. Although SSIM is traditionally used for reference-based comparison, in fusion tasks it is typically computed between the fused image and each source image separately. This approach allows assessment of the balance between thermal feature preservation and polarization-based structural information.

Perceptual Criteria

Perceptual metrics attempt to approximate characteristics of human visual perception. Visual Information Fidelity (VIF) and related measures estimate how much visually meaningful information is conveyed by the fused image [21]. In applications involving human operators, such as surveillance or visual inspection, perceptual criteria often correlate better with subjective image quality than purely statistical metrics.

In addition, subjective expert evaluation remains an important component of fusion assessment, particularly when the fused imagery is intended for direct human interpretation. Such evaluations consider scene readability, target discernibility, and the absence of visually disturbing artifacts [37].

Physically Motivated Criteria

For polarization–infrared fusion, physically motivated criteria are of particular importance. These metrics assess whether the fused image remains consistent with the underlying physical principles governing thermal emission and polarization. This includes preservation of thermal contrast associated with genuine temperature or emissivity differences, as well as faithful representation of polarization characteristics without artificial distortion.

Physically inconsistent enhancement of polarization features may lead to erroneous interpretation of surface material properties, which is unacceptable in scientific, medical, or defense-related applications. Therefore, some studies evaluate consistency with Stokes parameters or derived quantities such as DoLP to ensure physical plausibility [40].

Toward a Comprehensive Evaluation Framework

No single criterion is sufficient to fully characterize fusion quality. As a result, modern research increasingly adopts a multi-criteria evaluation framework, combining informational, structural, perceptual, and physical metrics [35–37]. This comprehensive approach enables more balanced and objective comparison of fusion algorithms, including classical transform-based methods and modern deep learning models.

Ultimately, fusion evaluation criteria serve not only as measurement tools but also as methodological instruments that shape algorithm design and guide future research directions by defining what constitutes “quality” in multimodal image fusion.

3.2 Application Domains

The fusion of polarization and infrared imagery has found increasing relevance across a wide spectrum of application domains, each of which imposes distinct requirements on fusion algorithms. The inherent complementarity of the two modalities – thermal radiometric contrast from IR sensors and structural, material, and surface-orientation cues from polarization measurements – enables enhanced scene understanding, target detection,

and material discrimination. The following subsections examine major domains where polarization–IR fusion has demonstrated substantial benefits [36][37].

Remote Sensing and Environmental Monitoring

In remote sensing applications, polarization–infrared fusion is particularly effective for environmental monitoring, terrain classification, and land–water boundary detection. Thermal infrared images provide robust temperature and emissivity information, which is relatively insensitive to changes in visible illumination or cloud cover. Polarization imagery, by contrast, captures surface texture, orientation, and scattering properties, which are essential for distinguishing between vegetation types, soil textures, and water surfaces.

For instance, studies have demonstrated that combining DoLP and AoP maps with LWIR images improves the detection of stressed vegetation, subtle changes in water turbidity, and geological features obscured by partial cloud cover [40]. Multi-resolution fusion methods allow analysts to extract both coarse thermal patterns and fine structural details simultaneously [17]. Hybrid methods, which incorporate both transform-based decompositions and adaptive weighting schemes, have been particularly effective in enhancing contrast between land cover classes while preserving spatial resolution [38].

Challenges in this domain include atmospheric interference, variable emissivity of natural materials, and sensor misregistration due to platform motion, which require careful preprocessing and calibration. Nevertheless, the integration of polarization information allows for a more nuanced interpretation of surface properties, enhancing the accuracy of classification and environmental assessment models.

Military and Surveillance Applications

Military and security operations represent one of the most prominent fields for polarization–infrared fusion. Low-visibility environments, including nighttime operations, fog, haze, and camouflage scenarios, often reduce the effectiveness of conventional optical sensors. Infrared imaging provides robust detection of heat-emitting targets, while polarization imaging reveals material properties and geometric structures that may be masked in intensity-only IR images [37].

For example, vehicles or personnel camouflaged against natural backgrounds may exhibit subtle polarization contrasts due to reflective surfaces or surface orientation changes, which are imperceptible in standard thermal imagery. Hybrid and learning-based fusion approaches enhance target detectability by selectively emphasizing salient thermal and polarization features while suppressing background clutter.

In addition, real-time fusion is critical for autonomous surveillance drones and ground vehicles. Lightweight CNN architectures and hybrid transform-learning models are increasingly employed to provide on-board processing that balances detection accuracy with computational efficiency. The combination of physical modeling, such as polarization consistency constraints, with adaptive learning frameworks ensures reliable target recognition in operationally challenging environments.

Biomedical Imaging and Diagnostics

Polarization–infrared fusion has emerged as a promising tool in biomedical imaging, where it contributes to non-invasive diagnostics and tissue characterization. Thermal infrared imaging can detect subtle temperature variations associated with vascular activity, inflammation, or tumor metabolism, while polarization parameters provide information about tissue anisotropy, fibrous structures, and surface morphology [41].

For instance, combining DoLP and AoP maps with thermal images enhances visualization of skin lesions, burns, or abnormal tissue growth by highlighting anisotropic scattering patterns that are invisible in standard IR images. Hybrid fusion techniques that integrate pixel-level weighting, multiresolution decomposition, and feature-level saliency measures have been shown to improve contrast between healthy and pathological tissue regions, potentially supporting early diagnosis of cancer or other conditions.

Challenges include motion artifacts, low polarization signal in soft tissues, and inter-subject variability in tissue properties. Deep learning approaches, particularly physics-informed networks, are increasingly used to address these challenges, learning robust fusion rules that preserve diagnostically relevant features while minimizing noise and artifact amplification.

Autonomous Navigation and Robotics

In autonomous navigation, including self-driving vehicles and robotic systems, reliable perception under adverse environmental conditions is paramount. Fog, dust, low-light, or nighttime scenarios can significantly reduce the effectiveness of standard visible-spectrum cameras. Infrared sensors provide thermal contrast for obstacle detection, while polarization imaging can reveal surface orientation, material properties, and boundaries that are otherwise hidden.

Fused polarization–IR imagery enables more robust obstacle recognition, terrain classification, and scene segmentation, which are critical for path planning and collision avoidance. Multi-level fusion strategies, combining pixel-, feature-, and decision-level integration, allow real-time systems to exploit both modalities efficiently. Transformer-based attention networks have recently been explored to dynamically prioritize salient regions, such as heated objects or high-contrast surfaces, improving the reliability of perception modules in autonomous systems.

Industrial and Cultural Heritage Applications

Beyond defense and biomedical domains, polarization–infrared fusion has applications in industrial inspection and cultural heritage preservation. In industrial settings, fused images can reveal defects, stress patterns, or thermal anomalies in machinery, electronic components, or composite materials. The polarization channel enhances detection of surface cracks, anisotropic textures, and material boundaries that may not manifest as thermal contrasts alone.

In cultural heritage preservation, combining infrared thermography with polarization imagery allows non-invasive examination of artworks, manuscripts, and architectural surfaces. Subsurface features, underdrawings, or layered materials can be revealed more effectively than with either modality alone, supporting restoration decisions and authenticity assessments.

These applications emphasize the versatility of fusion methods across domains with widely differing constraints on spatial resolution, temporal response, and interpretability.

Synthesis

Across all application domains, the complementary nature of polarization and infrared data provides a consistent advantage: the ability to extract thermal and structural/material information simultaneously. The choice of fusion methodology – classical, hybrid, or learning-based – depends on specific operational requirements, including real-time processing, interpretability, robustness to noise, and physical consistency. While classical and hybrid methods remain effective for well-defined and computationally constrained environments [38], deep learning approaches increasingly dominate domains requiring adaptive feature extraction and scene-dependent optimization.

Ultimately, the domain-driven analysis of fusion methods highlights that the design of fusion algorithms cannot be modality-agnostic; instead, it must account for both the physics of the sensors and the operational objectives of the target application.

4. EPISTEMOLOGICAL AND METHODOLOGICAL FOUNDATIONS OF POLARIZATION–INFRARED FUSION

Section 3 demonstrated the algorithmic evolution of polarization–infrared image fusion from transform-domain techniques to deep learning and physics-informed architectures. While those developments describe how fusion is implemented computationally, a deeper

question concerns how multimodal physical information is conceptually integrated. This section examines the epistemological and methodological foundations underlying polarization–infrared fusion, providing the theoretical context necessary for understanding the application-driven validation discussed in Section 5.

4.1. Classical Analytical Paradigm: Fusion as Signal Aggregation

Early polarization–infrared fusion methods emerged within a classical signal-processing paradigm [14, 42, 43]. In this framework, each modality is treated as an independent measurement channel. Infrared radiance and polarization descriptors (e.g., DoLP, AoLP, Stokes parameters) are considered separate scalar or vector fields representing distinct physical observables.

Transform-domain approaches – such as Laplacian pyramids, wavelet transforms, and multiscale decomposition – map these modalities into a shared frequency representation, where fusion rules select coefficients based on predefined criteria (e.g., maximum absolute value, local variance, or energy preservation) [16, 41]. The epistemological assumption underlying these methods is additive complementarity: each modality contributes partial but compatible information that can be combined through deterministic rules.

However, infrared and polarization imaging originate from fundamentally different physical processes:

- Infrared radiance obeys Planck's law and thermodynamic emission constraints.
- Polarization contrast arises from boundary conditions in electromagnetic wave reflection and emission governed by Fresnel relations.

The analytical paradigm treats these outputs as statistical signals without explicitly modeling their shared physical causality. Although effective for enhancement tasks, this abstraction limits interpretability and may produce physically inconsistent fused outputs under complex environmental conditions.

4.2. Systems-Oriented Integration: From Independence to Interaction

As discussed in Section 3, limitations of fixed-rule methods motivated adaptive and model-based strategies. These developments reflect a methodological shift from reductionism to systems-oriented integration [44].

In real scenes:

- Surface emissivity affects both thermal intensity and polarization state.
- Viewing geometry modifies DoLP and apparent thermal contrast.
- Atmospheric scattering influences polarization differently from radiometric attenuation.

Thus, modalities are not independent; they are coupled through shared environmental and material parameters.

Systems-oriented fusion models treat polarization and infrared signals as outputs of a unified physical scene rather than independent inputs. Hybrid approaches began incorporating:

1. Scene-adaptive weighting,
2. Region-based feature selection,
3. Physical parameter modeling (e.g., emissivity–angle relations),
4. Statistical dependence analysis.

In this framework, fusion is relational rather than additive. Information emerges from cross-modal interaction under shared constraints. This methodological transformation laid the groundwork for learning-based integration.

4.3. Computational Mediation: Representation in Deep Fusion Models

The introduction of convolutional neural networks (CNNs) and transformer-based architectures marked a significant transition in fusion research [18, 46, 47]. Unlike

deterministic transform methods, deep networks construct hierarchical feature representations.

Typical architectures consist of:

1. Modality-specific encoders,
2. Cross-modal fusion modules,
3. Reconstruction or decision layers.

In polarization–infrared fusion, learned intermediate representations capture:

- Spatial gradients,
- Thermal contrast distributions,
- Polarization anisotropy patterns,
- Cross-channel correlations.

From an epistemological standpoint, this introduces computational mediation: knowledge is inferred through learned feature spaces rather than predefined physical rules. The fusion mechanism becomes data-driven and optimization-based.

However, purely empirical learning introduces methodological concerns:

- Latent features lack explicit physical interpretability.
- Generalization across environmental conditions may degrade.
- Physical constraints (e.g., DoLP ≤ 1) are not automatically enforced.

These issues highlight the need to reconcile inductive learning with deductive physical modeling.

4.4. Physics-Informed Fusion: Integrating Law and Learning

Recent research trends, partially introduced in Section 3, address interpretability and robustness through physics-informed modeling [48]. These methods embed physical constraints directly into network architecture or loss functions.

Examples include:

- Enforcing valid Stokes parameter relationships,
- Penalizing nonphysical polarization magnitudes,
- Preserving radiometric consistency,
- Maintaining angular continuity of polarization vectors.

Physics-informed loss functions constrain optimization within physically plausible solution spaces. This restores theoretical grounding without sacrificing adaptive capability.

Methodologically, physics-informed fusion represents a synthesis of two epistemic traditions:

- Deductive reasoning based on electromagnetic and thermodynamic theory.
- Inductive inference through data-driven optimization.

Rather than replacing analytical modeling, deep learning becomes a flexible approximator operating within physical law constraints. In this context, Jones-formalism-based models demonstrate how classical polarization optics can be integrated with modern computational fusion strategies, establishing a bridge between physically grounded analytical representations and adaptive learning-based architectures [30].

4.5. Comparative Conceptual Framework

To clarify the methodological progression described above, a comparative conceptual framework is proposed in **Table 2**.

The diagram visually demonstrates the transition from additive signal combination to constraint-aware integrative systems.

Table 2. **Comparative Methodological Evolution of Polarization–Infrared Fusion**

Component	Classical Analytical Model	Data-Driven Deep Fusion	Physics-Informed Hybrid Model
Inputs	Infrared intensity Polarization parameters	Raw infrared image Raw polarization image	Infrared radiance Stokes polarization parameters Physical priors
Processing Approach	Transform-domain decomposition (wavelet, pyramid, or multiscale transforms) Fixed fusion rule (e.g., coefficient selection or weighted averaging)	CNN or Transformer encoders for feature extraction Feature-level fusion layers Reconstruction or decoding network	Neural feature extraction module Physics-constrained fusion mechanism Regularized loss function incorporating physical priors
Output	Enhanced composite image	Optimized fused image	Physically consistent fused representation
Adaptivity	Low – relies on predefined rules	High – model learns fusion strategy from data	High – adaptive learning guided by physical constraints
Interpre-tability	High – processing steps mathematically transparent	Limited – internal representations often opaque	Moderate to high – interpretability supported by physical modeling
Physical Consistency	Implicit – physical relationships not explicitly modeled	Low – fusion driven primarily by data patterns	High–fusion constrained by radiometric and polarization physics
Performance Characte-ristics	Stable and computationally efficient, but limited in complex scenes	High performance in diverse scenarios, but potentially sensitive to dataset bias	Balanced performance combining generalization capability with physical reliability

4.6. Evaluation as an Epistemic Criterion

As fusion methodologies evolved, evaluation strategies also transformed. Metrics such as entropy, mutual information, and the structural similarity index measure (SSIM) have traditionally been used to quantify information preservation and structural coherence.

However, these metrics encode implicit normative assumptions:

- Entropy prioritizes diversity.
- Mutual information prioritizes cross-modal retention.
- SSIM prioritizes spatial structure.

For polarization–infrared fusion, a physically meaningful evaluation must additionally consider:

- Preservation of genuine thermal gradients,
- Integrity of polarization signatures,
- Absence of artificial enhancement artifacts,
- Radiometric consistency.

Thus, evaluation functions not merely as a performance measurement but as an epistemological filter shaping methodological development. Algorithms optimized under specific metrics implicitly define what constitutes a “good” fused image.

4.7. Interdisciplinary Context

Polarization–infrared fusion integrates principles from:

- Physical optics,
- Electromagnetic theory,
- Signal processing,
- Statistical learning,
- Systems engineering.

Each discipline contributes constraints and modeling strategies. Effective fusion requires harmonizing theoretical rigor with computational adaptability. Overreliance on purely empirical optimization risks sacrificing interpretability, while strict analytical modeling may limit robustness in complex scenes.

This interdisciplinary synthesis prepares the transition to Section 5, where operational performance and application-driven validation are examined.

4.8. Transition to Applications

The methodological developments outlined above directly influence practical performance in:

Target detection under low contrast,
Camouflage discrimination,
Maritime and aerial surveillance,
Autonomous navigation,
Biomedical imaging diagnostics.

Physics-informed and system-oriented models demonstrate improved robustness under environmental variability and noise – factors critical in real-world deployment.

Section 5 therefore, evaluates fusion algorithms not solely by quantitative metrics but by operational reliability and application-specific effectiveness. The epistemological framework developed in this section provides the theoretical foundation for interpreting those results.

5. APPLICATION-DRIVEN VALIDATION AND OPERATIONAL PERFORMANCE OF POLARIZATION–INFRARED FUSION

Section 3 analyzed algorithmic evolution, and Section 4 established the epistemological and methodological foundations of polarization–infrared image fusion. The present section evaluates how these theoretical and computational developments translate into operational performance across real-world application domains. Fusion quality must ultimately be validated not only through numerical metrics but through task-oriented reliability under environmental variability, sensor noise, and scene complexity.

Polarization–infrared fusion is particularly valuable in conditions where visible imaging fails or provides insufficient discriminative information. Its principal operational strength lies in combining:

- Thermal contrast (temperature-dependent radiative emission),
- Polarimetric contrast (surface geometry and material-dependent polarization signatures).

This dual sensitivity enhances target discrimination, material classification, and structural interpretation.

5.1. Target Detection and Surveillance

One of the primary application domains of infrared imaging is target detection under low illumination or adverse weather conditions. However, thermal contrast alone may be insufficient when temperature differences between target and background are small or dynamically varying [48].

Polarization imaging provides complementary information, particularly for detecting:

- Man-made objects against natural backgrounds,
- Surface discontinuities,
- Camouflaged structures,
- Specular reflections from engineered materials [10].

Studies have shown that man-made objects often exhibit stronger or more structured polarization signatures compared to natural vegetation or soil [49]. When fused with infrared radiance, polarization features improve:

- Signal-to-clutter ratio,
- Edge clarity,
- False alarm reduction.

Deep fusion architectures further enhance performance by learning discriminative cross-modal features optimized for detection objectives [17].

Operational validation typically involves:

- Receiver operating characteristic (ROC) analysis,
- Probability of detection vs. false alarm rate,
- Target-background separability metrics.

Physics-informed fusion models demonstrate improved robustness in cases where polarization contrast varies with viewing geometry, as physical constraints stabilize representation [18].

5.2. Camouflage and Concealment Discrimination

Camouflage detection represents a critical use case where fusion provides clear advantages. Thermal camouflage attempts to match background temperature, reducing infrared contrast. However, matching polarization signatures is substantially more difficult due to dependence on:

- Surface microstructure,
- Reflectance anisotropy,
- Material refractive index.

Polarimetric contrast remains detectable even when thermal gradients are minimized. Fusion strategies amplify subtle inconsistencies between thermal and polarization responses.

Empirical studies demonstrate that fused imagery increases classification accuracy in camouflage scenarios by integrating:

- Local polarization anisotropy,
- Thermal spatial gradients,
- Texture features.

Machine learning-based fusion frameworks outperform fixed-rule approaches when background complexity increases. However, physics-informed regularization prevents over-amplification of noise in polarization channels, which are often more sensitive to environmental variability.

5.3. Maritime and Aerial Remote Sensing

Maritime environments present challenging conditions:

- Low thermal contrast between vessels and water,
- High specular reflection,
- Dynamic atmospheric scattering.

Polarization is particularly effective in water–air boundary detection and glare suppression. Infrared imaging enhances the detection of thermal signatures from engines or exhaust systems.

Fusion in maritime surveillance improves:

- Vessel boundary delineation,
- Wake detection,

- Oil spill discrimination,
- Floating object identification [48].

In aerial remote sensing, polarization assists in distinguishing man-made infrastructure from natural terrain. Infrared channels contribute to temperature-based material separation.

Operational studies show that multiscale fusion enhances spatial resolution and reduces clutter in coastal monitoring scenarios [45]. Transformer-based models have demonstrated improved generalization across environmental conditions compared to conventional CNN-based architectures.

5.4. Autonomous Navigation and Robotics

Autonomous systems require robust perception under variable illumination, weather, and terrain conditions. Infrared imaging ensures functionality in low-light scenarios, while polarization assists in:

- Road surface detection,
- Ice or water hazard identification,
- Surface orientation estimation [46].

Fusion contributes to improved edge detection and structural recognition in outdoor navigation tasks. Cross-modal attention mechanisms allow dynamic weighting depending on environmental context.

In robotics, fused polarization–infrared inputs improve object segmentation and obstacle detection, particularly in low-contrast environments [50].

Real-time constraints impose additional requirements:

- Computational efficiency,
- Hardware compatibility,
- Low latency.

Physics-informed lightweight networks demonstrate promising trade-offs between interpretability and speed for embedded systems.

5.5. Biomedical Imaging Applications

Beyond defense and surveillance, polarization–infrared fusion shows potential in biomedical diagnostics. Infrared imaging captures temperature variations related to inflammation or vascular abnormalities. Polarization imaging reveals structural changes in tissue microarchitecture [47]

Applications include:

- Skin lesion detection,
- Burn assessment,
- Tumor boundary visualization,
- Vascular mapping.

Fusion enhances diagnostic reliability by integrating thermal anomalies with polarization-sensitive structural contrast. Clinical studies suggest improved sensitivity in detecting subsurface irregularities compared to single-modality imaging [51].

However, biomedical validation requires strict radiometric calibration and careful handling of polarization noise, as biological tissues introduce complex scattering behavior.

5.6. Evaluation Metrics and Operational Criteria

While classical metrics such as entropy, mutual information, and SSIM remain widely used, application-driven validation emphasizes task-specific criteria:

- Detection accuracy,
- Segmentation precision,
- Classification reliability,
- Robustness to noise,

- Stability across viewing angles.
- Target visibility [31].

Recent research increasingly incorporates:

- Task-driven loss functions,
- End-to-end detection pipelines,
- Real-time benchmarking.

Physics-consistent fusion models exhibit superior stability under environmental perturbations compared to unconstrained networks, particularly when polarization signal-to-noise ratio is low [52].

Additionally, uncertainty quantification methods are emerging to estimate confidence in fused outputs – an important factor for safety-critical systems [53].

5.7. Limitations and Practical Challenges

Despite demonstrated advantages, polarization–infrared fusion faces several practical limitations:

1. Sensor Calibration Complexity

Accurate alignment of polarization channels and infrared radiometry requires precise calibration.

2. Polarization Noise Sensitivity

Polarimetric measurements are sensitive to atmospheric scattering and sensor misalignment.

3. Dataset Scarcity

Public datasets containing synchronized, calibrated polarization–infrared pairs remain limited.

4. Ground Truth Ambiguity

Defining objective ground truth for fused imagery is inherently challenging.

Addressing these challenges requires standardized benchmarks and shared evaluation protocols.

5.8. Synthesis and Transition

Application-driven validation confirms that the methodological principles described in Sections 3 and 4 directly influence operational robustness. Systems-oriented and physics-informed approaches demonstrate enhanced stability, interpretability, and cross-domain generalization.

Fusion has evolved from visual enhancement toward operational perception support. In contemporary systems, fused imagery often feeds higher-level modules:

- Object detection networks,
- Tracking systems,
- Decision-support algorithms.

Thus, polarization–infrared fusion becomes an integral component of multimodal perception architectures rather than an isolated preprocessing step.

CONCLUSION

The review of image-fusion techniques using polarization and infrared intensity reveals a trajectory that mirrors the broader evolution of scientific thought: from analytical separation to systemic integration. Technically, this progression moves from transform-based and energy-driven methods toward adaptive, learning-based architectures that embody physical and cognitive principles simultaneously. Conceptually, it demonstrates the convergence of physics, computation, and epistemology in the modern understanding of perception and information.

Future work will require:

- development of standardized multi-modal datasets;

- creation of interpretable, physics-aware neural architectures;
- and exploration of cognitive models that explain how fusion contributes to understanding rather than mere visualization.

Ultimately, polarization–infrared image fusion exemplifies the post-non-classical vision of science – where observation, computation, and cognition become integrated aspects of a single evolving system of knowledge.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [V.A., O.K.]; methodology, [V.A., O.K.]; writing – original draft preparation, [V.A.]; writing – review and editing, [O.K.]; visualization, [V.A.], supervision, [O.K.];

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Ma, W., Wang, K., Li, J., Yang, S. X., Song, L., & Li, Q. (2023). Infrared and visible image fusion technology and application: A review. *Sensors*, *23*(2), 599. https://doi.org/10.3390/s23020599

[2] Kolobrodov, V. G. (2024). Energy model of optical system of polarimetric thermal imager. *Visnyk NTUU KPI Seriia - Radiotekhnika Radioaparatobuduvannia*, *95*, 47–53 (in Ukrainian). https://doi.org/10.20535/RADAP.2024.95.47-53

[3] Guo, F., Zhu, J., Huang, L., Li, F., Zhang, N., Deng, J., Li, H., Zhang, X., Zhao, Y., & Jiang, H. (2024). Multi-dimensional fusion of spectral and polarimetric images followed by pseudo-color algorithm integration and mapping in HSI space. *Remote Sensing*, *16*(7), 1119. https://doi.org/10.3390/rs16071119

[4] A.-J. Fresnel (1821). Mémoire sur les modifications que la réflexion imprime à la lumière polarisée. *Annales de Chimie et de Physique*, *17*, 45-176. https://doi.org/10.5281/ZENODO.5060346

[5] Stokes, G. G. (1851). On the composition and resolution of streams of polarized light from different sources. *Transactions of the Cambridge Philosophical Society*, *9*, 399. https://doi.org/10.1017/CBO9780511702266.010

[6] Born M, Wolf E, Bhatia AB, et al. (1999). *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*. 7th ed. Cambridge University Press. https://doi.org/10.1017/CBO9781139644181

[7] Herschel, W. (1800). II. Experiments on the refrangibility of the invisible rays of the sun. *The Philosophical Magazine*, *8*(29), 9-15. https://doi.org/10.1080/14786440008562602

[8] Planck, M. (1901). On the law of distribution of energy in the normal spectrum. *Annalen der physik*, *4*, 553-563. https://doi.org/10.1002/andp.19013090310

[9] Jones, R. C. (1941). A new calculus for the treatment of optical systems I. Description and discussion of the calculus. *Journal of the Optical Society of America*, *31*(7), 488-493. https://doi.org/10.1364/JOSA.31.000488

[10] Rogalski, A. (2003). Infrared detectors: status and trends. *Progress in quantum electronics*, *27*(2-3), 59-210. https://doi.org/10.1016/S0079-6727(02)00024-1
[11] Rogalski, A. (2010). *Infrared Detectors* (2nd ed.). CRC Press. https://doi.org/10.1201/b10319
[12] Fossum, E. R. (1997). CMOS image sensors: Electronic camera-on-a-chip. *IEEE transactions on electron devices*, *44*(10), 1689-1698. https://doi.org/10.1109/16.628824
[13] Gruev, V., Ortu, A., Yang, Z., Van der Spiegel, J., & Engheta, N. (2007, September). High-resolution integrated image sensor with polymer micropolarization array. In *Frontiers in Optics* (p. FTuS6). Optica Publishing Group. https://doi.org/10.1364/FIO.2007.FTuS6
[14] Burt, P. J., & Adelson, E. H. (1983). The Laplacian Pyramid as a Compact Image Code. *IEEE Transactions on Communications*, *31*(4), 532-540. https://doi.org/10.1109/TCOM.1983.1095851
[15] Mallat, S. G. (2002). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, *11*(7), 674-693. https://doi.org/10.1109/34.192463
[16] Piella, G. (2003). A general framework for multiresolution image fusion: from pixels to regions. *Information fusion*, *4*(4), 259-280. https://doi.org/10.1016/S1566-2535(03)00046-0
[17] Tyo, J. S., Goldstein, D. L., Chenault, D. B., & Shaw, J. A. (2006). Review of passive imaging polarimetry for remote sensing applications. *Applied optics*, *45*(22), 5453-5469. https://doi.org/10.1364/AO.45.005453
[18] Li, H., & Wu, X. J. (2018). DenseFuse: A fusion approach to infrared and visible images. *IEEE transactions on image processing*, *28*(5), 2614-2623. https://doi.org/10.1109/TIP.2018.2887342
[19] aswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*, 5998-6008. https://doi.org/10.48550/arXiv.1706.03762
[20] Yang, J., Fu, X., Hu, Y., Huang, Y., Ding, X., & Paisley, J. (2017). PanNet: A deep network architecture for pan-sharpening. In *Proceedings of the IEEE international conference on computer vision* (pp. 5449-5457). https://doi.org/10.1109/ICCV.2017.193
[21] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, *13*(4), 600-612. https://doi.org/10.1109/TIP.2003.819861
[22] Zhang, Q., & Guo, B. L. (2009). Multifocus image fusion using the nonsubsampled contourlet transform. *Signal processing*, *89*(7), 1334-1346. https://doi.org/10.1016/j.sigpro.2009.01.012
[23] Pratt, W. K. (2007). *Digital image processing: PIKS Scientific inside* (Vol. 4). Hoboken, New Jersey: Wiley-Interscience. https://doi.org/10.1002/0470097434
[24] Yue, Z., & Li, F. M. (2014, February). An infrared polarization image fusion algorithm based on oriented Laplacian pyramid. In *Selected Papers from Conferences of the Photoelectronic Technology Committee of the Chinese Society of Astronautics: Optical Imaging, Remote Sensing, and Laser-Matter Interaction 2013* (Vol. 9142, pp. 60-70). SPIE. https://doi.org/10.1117/12.2054074
[25] Zhang, J., Zhang, Y., & Shi, Z. (2018). Long-wave infrared polarization feature extraction and image fusion based on the orthogonality difference method. *Journal of Electronic Imaging*, *27*(2), 023021-023021. https://doi.org/10.1117/1.JEI.27.2.023021

[26] Cunha, A. L., Zhou, J., & Do, M. N. (2006). The nonsubsampled contourlet transform: Theory, design, and applications. *IEEE Transactions on Image Processing*, *15*(10), 3089–3101. https://doi.org/10.1109/TIP.2006.877507
[27] Khaustov, Ya. Ye., Khaustov, D. Ye., Lychkovskyy, E., Ryzhov, Ye., & Nastishin, Yu. A. (2019). Image fusion for a target sightseeing system of armored vehicles. *Milit. Techn. Collect.*, *21*, 28-37. https://doi.org/10.33577/2312-4458.21.2019.28-37
[28] Khaustov, Ya. Ye., Khaustov, D. Ye., Ryzhov, Ye., Lychkovskyy, E., & Nastishin, Yu. A. (2020). Fusion of visible and infrared images via complex function. *Milit. Techn. Collect.*, *22*, 20-31. https://doi.org/10.33577/2312-4458.22.2020.20-31
[29] Khaustov, Ya. Ye., Khaustov, D. Ye., Hryvachevskyi, A. P., Ryzhov, Ye., Lychkovskyy, E., Prudyus, I. N., & Nastishin, Yu. A. (2021). Complex function as a template for image fusion. *Results in Optics*, 2, 100038. https://doi.org/10.1016/j.rio.2020.100038
[30] Khaustov, D. Ye., Khaustov, Ya. Ye., Ryzhov, Ye., Lychkovskyy, E., Vlokh, R., & Nastishin, Yu. A. (2021). Jones formalism for image fusion. *Ukr. J. Phys. Opt.*, *22*(3), 165-180. https://doi.org/10.3116/16091833/22/3/165/2021
[31] Khaustov, D. Ye.,Kyrychuk, O. A., Khaustov, Ya. Ye., Stakh, Ya. Ye., Zhyrna O. V., & Nastishin, Yu. A. (2023). The Measure of Target Saliency for Target-Oriented Image Fusion. *Collection of scientific papers of the State Research Institute for Testing and Certification of Armaments and Military Equipment*, *3*(17), 122-136. https://doi.org/10.37701/dndivsovt.17.2023.15
[32] Khaustov, D., Khaustov, Ya., Ryzhov, Ye., Burashnikov, O., Lychkovskyy, E., & Nastishin, Yu. (2021). Dynamic fusion of images from the visible and infrared channels of sightseeing system by complex matrix formalism. *Milit. Techn. Collect.*, *25*, 29-37. https://doi.org/10.33577/2312-4458.25.2021.29-37
[33] Khaustov D. Ye., Kyrychuk O. A., Stakh T. M., Khaustov Ya. Ye., Burashnikov O. O., Ryzhov Ye., Vlokh R., & Nastishin Yu. A. (2023). Complex-scalar and complex-vector approaches for express target-oriented image fusion. *Ukr. J. Phys. Opt.*, *24*(1), 62-82. https://doi.org/10.3116/16091833/24/1/62/2023
[34] Li, K., Qi, M., Zhuang, S., Liu, Y., & Gao, J. (2023). Noise-aware infrared polarization image fusion based on salient prior with attention-guided filtering network. *Optics Express*, *31*(16), 25781-25796. https://doi.org/10.1364/OE.492954
[35] Duan, J., Zhang, H., Liu, J., Gao, M., Cheng, C., & Chen, G. (2023). A dual-weighted polarization image fusion method based on quality assessment and attention mechanisms. *Frontiers in Physics*, *11*, 1214206. https://doi.org/10.3389/fphy.2023.1214206
[36] Jagalingam, P., & Hegde, A. V. (2015). A review of quality metrics for fused image. *Aquatic Procedia*, *4*, 133-142. https://doi.org/10.1016/j.aqpro.2015.02.019
[37] Kaur, H., Koundal, D., & Kadyan, V. (2021). Image Fusion Techniques: A Survey. *Archives of computational methods in Engineering*, *28*(7), 4425-4447. https://doi.org/10.1007/s11831-021-09540-7
[38] Jin, X., Jiang, Q., Yao, S., Zhou, D., Nie, R., Hai, J., & He, K. (2017). A survey of infrared and visual image fusion methods. *Infrared Physics & Technology*, *85*, 478-501. https://doi.org/10.1016/j.infrared.2017.07.010
[39] Zhang, H., Xu, H., Tian, X., Jiang, J., & Ma, J. (2021). Image fusion meets deep learning: A survey and perspective. *Information Fusion*, *76*, 323-336. https://doi.org/10.1016/j.inffus.2021.06.008
[40] Sheikh, H. R., & Bovik, A. C. (2006). Image information and visual quality. *IEEE Transactions on Image Processing*, *15*(2), 430-444. https://doi.org/10.1109/TIP.2005.859378

[41] Zhang, Z., & Blum, R. S. (1999). A categorization of multiscale-decomposition-based image fusion schemes with a performance study for a digital camera application. *Proceedings of the IEEE*, *87*(8), 1315-1326. https://doi.org/10.1109/5.775414
[42] Wang, L., Dou, J., Qin, P., Lin, S., Gao, Y., Wang, R., & Zhang, J. (2021). Multimodal medical image fusion based on nonsubsampled shearlet transform and convolutional sparse representation. *Multimedia Tools and Applications*, *80*(30), 36401-36421. https://doi.org/10.1007/s11042-021-11379-w
[43] Li, H., Manjunath, B. S., & Mitra, S. K. (1995). Multisensor image fusion using the wavelet transform. *Graphical models and image processing*, *57*(3), 235-245. https://doi.org/10.1006/gmip.1995.1022
[44] Li, S., Kang, X., & Hu, J. (2013). Image fusion with guided filtering. *IEEE Transactions on Image Processing*, *22*(7), 2864-2875. https://doi.org/10.1109/TIP.2013.2244222
[45] Wald, L. (1999). Some terms of reference in data fusion. *IEEE Transactions on Geoscience and Remote Sensing*, *37*(3), 1190-1193. https://doi.org/10.1109/36.763269
[46] Ma, J., Ma, Y., & Li, C. (2019). Infrared and visible image fusion methods and applications: A survey. *Information fusion*, *45*, 153-178. https://doi.org/10.1016/j.inffus.2018.02.004
[47] Xu, H., Ma, J., Jiang, J., Guo, X., & Ling, H. (2020). U2Fusion: A unified unsupervised image fusion network. *IEEE transactions on pattern analysis and machine intelligence*, *44*(1), 502-518. https://doi.org/10.1109/TPAMI.2020.3012548
[48] Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, *378*, 686-707. https://doi.org/10.1016/j.jcp.2018.10.045
[49] Goldstein, D.H. (2011). *Polarized Light* (3rd ed.). CRC Press. https://doi.org/10.1201/b10436
[50] Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, *32*(11), 1231-1237. https://doi.org/10.1177/0278364913491297
[51] Tuchin, V. V. (2015). *Tissue Optics: Light Scattering Methods and Instruments for Medical Diagnostics,* Third Edition. Society of Photo-Optical Instrumentation Engineers (SPIE). https://doi.org/10.1117/3.1003040
[52] Drgoňa, J., Tuor, A. R., Chandan, V., & Vrabie, D. L. (2021). Physics-constrained deep learning of multi-zone building thermal dynamics. *Energy and Buildings*, *243*, 110992. https://doi.org/10.1016/j.enbuild.2021.110992
[53] Kendall, A., & Gal, Y. (2017). What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? *arXiv preprint arXiv:1703.04977*. https://doi.org/10.48550/arXiv.1703.04977

МЕТОДИКИ КОМПЛЕКСУВАННЯ ЗОБРАЖЕНЬ З ВИКОРИСТАННЯМ ПОЛЯРИЗАЦІЙНОЇ ІНФОРМАЦІЇ ТА ІНТЕНСИВНОСТІ ІНФРАЧЕРВОНОГО ВИПРОМІНЮВАННЯ

Віталій Артим*, Олег Крупич
Кафедра оптоелектроніки та інформаційних технологій,
Львівський національний університет імені Івана Франка,
вул. Ген. Тарнавського 107, Львів, 79017, Україна

АНОТАЦІЯ

Вступ. Комплексування поляризаційних та звичайних теплових інфрачервоних зображень поєднує взаємодоповнювальну фізичну інформацію для покращення інтерпретації сцени. Інфрачервоне інтенсивнісне зображення відображає теплове випромінювання та розподіл коефіцієнта випромінювальної здатності, тоді як параметри поляризації, зокрема ступінь і кут лінійної поляризації, кодують мікроструктуру поверхні, матеріальний склад і геометричну орієнтацію, що є недоступними для сенсорів, заснованих лише на інтенсивності. Їхнє комплексування підвищує просторову деталізацію, розпізнавання об'єктів і стійкість за умов погіршеної видимості, відображаючи ширший перехід до мультимодального сприйняття.

Матеріали та методи. У роботі подано аналітичний огляд методів комплексування поляризаційних та інфрачервоних зображень. Розвиток підходів розглянуто від класичних лінійних перетворень, багатомасштабної декомпозиції та енергетичних правил прийняття рішень до адаптивних гібридних схем. Особливу увагу приділено архітектурам глибокого навчання та фізично-інформованим моделям, що враховують принципи формування поляризаційного сигналу та радіометричні обмеження. Прогрес у сфері комплексування проаналізовано в контексті розвитку сенсорних технологій і обчислювальної візуалізації.

Результати. Аналіз свідчить, що включення поляризаційної складової суттєво підвищує інформативність синтезованих зображень порівняно з традиційними інтенсивнісними методами. Класичні алгоритми забезпечують стабільні та інтерпретовані результати, проте мають обмежену адаптивність у складних умовах. Підходи на основі навчання та гібридні стратегії демонструють кращу збереженість контрасту, покращену структурну деталізацію та вищу виявлюваність цілей, особливо в умовах теплової неоднорідності або візуальної деградації. Серед актуальних проблем – обмеженість анотованих наборів даних, чутливість до поляризаційного шуму та недостатня фізична інтерпретованість глибинних нейронних моделей.

Висновки. Комплексування поляризаційних та інфрачервоних зображень відображає перехід від одномодального до багатомодального спостереження для системно-орієнтованого сприйняття, що поєднує фізичне моделювання з підходами на основі даних. Подальші дослідження мають бути зосереджені на фізично-обґрунтованому проєктуванні нейронних архітектур, стандартизації протоколів оцінювання та підвищенні інтерпретованості для забезпечення надійного практичного впровадження.

Ключові слова: комплексування зображень, поляризаційне зображення, інфрачервоне зображення, мультимодальні дані, глибоке навчання

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
18 May, 2026	03 June, 2026	05 June, 2026	29 June, 2026

Electronics and information technologies, 2026, 34,31–50
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC 004.94

MULTI-PARAMETER DYNAMIC LEARNING MAPS OF NEURAL NETWORKS TRAINED WITH ADAM OPTIMISER

***Sergiy Sveleba*[1]*, *Ivan Katerynchuk*[1], *Yaroslav Shmyhelskyy*[1], *Lucjan Pelc*[2], *Volodymyr Brygilevych*[2], *Nazar Sveleba*[3]**
[1]*Ivan Franko National University of Lviv,*
107 Gen. Tarnavskoho Str., Lviv, 79017, Ukraine
[2]*The State Higher School of Technology and Economics in Jarosław,*
ul. Czarnieckiego 16, Jarosław, 37-500, Poland
[3]*Private Higher Education Establishment "European University",*
16B, Academician Vernadsky Boulevard, Kyiv, Ukraine

Sveleba, S., Katerynchuk, I., Shmyhelskyy, Ya., Pelc, L., Brygilevych, V., Sveleba, N. (2026). Multi-Parameter Dynamic Learning Maps of Neural Networks Trained with Adam Optimizer. *Electronics and Information Technologies*, 34, 31-50. https://doi.org/10.30970/eli.34.2

ABSTRACT

Background. Understanding how hyperparameters of the Adam optimizer shape the learning dynamics of multilayer neural networks remains an open problem, particularly in regimes where training transitions from stable to chaotic behavior. Previous studies have shown that learning rate and network complexity can lead to underfitting, efficient learning, overfitting, or chaotic states; however, systematic visualization of these regimes in the β_1–β_2 parameter space is still lacking.

Materials and Methods. Dynamic regime maps are constructed for a multilayer neural network trained on binary-encoded digit patterns. The training process is analyzed using a recurrent two-dimensional mapping based on the error evolution of a neuron and its neighbor. Periodicities in the final iterations are identified via the convergence of periodicities method, enabling classification of learning regimes from fixed points to high-order cycles and chaos. The maps are generated for varying learning rates α, training duration, network depth, and neuron count, with β_1 and β_2 scanned over the interval [0, 0.999].

Results and Discussion. The results demonstrate strong sensitivity of Adam dynamics to the joint values of β_1 and β_2. Small learning rates ($\alpha = 0.001$) produce predominantly first-order periodicity, indicating underfitting. Moderate values ($\alpha = 0.1 - 0.4$) reveal structured transitions from rapid to slower learning as β_2 increases. In contrast, large α (≥ 0.6) leads to fragmented, non-monotonic patterns associated with overfitting and chaotic dynamics. Increasing network size introduces noisy gradients, shrinking stable regions and expanding chaotic zones, whereas smaller networks yield smoother regime transitions.

Conclusion. Dynamic regime mapping in the β_1–β_2 space serves as an effective diagnostic tool for identifying stable, inefficient, and chaotic learning regimes. The findings confirm the high sensitivity of Adam to hyperparameter interactions and network architecture, showing that inappropriate combinations can induce chaotic behavior even at moderate learning rates. The proposed framework provides a novel approach for analyzing optimizer dynamics and guiding hyperparameter selection in deep learning.

Keywords: Adam optimizer; learning dynamics; periodicity maps; hyperparameter tuning; chaotic learning; multilayer neural networks.

The present study should primarily be interpreted as a qualitative analysis of the dynamic properties of neural-network learning under controlled conditions, rather than as a benchmark comparison on large-scale datasets. The use of a binary 4×7 dataset was intentionally chosen to minimize the influence of high-dimensional factors and to improve the interpretability of optimizer dynamics. Preliminary experiments on MNIST and CIFAR-10 datasets demonstrated qualitatively similar fragmentation of stability regions, although with significantly higher noise levels and lower interpretability.

INTRODUCTION

Optimizers adjust network parameters to minimize a loss function and are central to training efficiency and generalization. Adam combines momentum on first moments with adaptive scaling by second moments, providing fast convergence in many settings but exhibiting sensitivity to the learning rate and moment decay factors (β_1, β_2). Here, we investigate how Adam's hyperparameters, together with iteration budget and model size, shape the qualitative dynamics of learning.

Background and related work

Neural network training optimizers are used to adjust model weights to minimize the loss function. They play a key role in the learning process, helping the network find the best parameter values to summarize the data and achieve high accuracy on the test set.

The main functions of optimizers can be considered:

- convergence acceleration – optimizers help to quickly find the minimum loss function;
- overcoming local minimums – optimizers that allow you to get out of local minimums and find better solutions;
- adaptability to different parameters – optimizers change the learning speed for different weights, improving the stability of learning;
- prevention of overtraining – optimizers or their modifications, which can contribute to better generalization of the model.

The main types of optimizers include gradient descent and adaptive methods.

Gradient Descent optimizers include:

- regular (Batch Gradient Descent), which uses the entire dataset to update the weights;
- stochastic (SGD – Stochastic Gradient Descent), which updates the weights after each cycle, which can speed up learning, but makes it unstable;
- minibatch (minibatch Gradient Descent), which is a compromise between the two previous methods.

Adaptive methods:

- Momentum – adds "inertia" to updates, helping to overcome local lows;
- Adagrad – adaptively changes the learning speed for different parameters;
- RMSprop is a modification of Adagrad that eliminates its main drawback – a decrease in the learning speed too quickly;
- Adam (Adaptive Moment Estimation) – combines the advantages of Momentum and RMSprop, one of the most popular optimizers;
- AdamW is an improved version of Adam with better regulation.

Each of the optimizers has its own advantages and is used depending on the task, the type of neural network, and the size of the data.

Adam is one of the most popular and effective optimizers for training neural networks. At its core, the Adam optimizer is an extension of stochastic gradient descent methods. It includes adaptive learning speed for each parameter, and dynamic step size adjustment during training. This adaptability allows the method to navigate

efficiently in an optimization environment, mitigating the problems associated with fixed learning rates. The algorithm's adaptability and momentum methods contribute to faster convergence and improved overall performance of deep neural networks [1]. Adam optimizer works as an extension of the traditional gradient descent algorithm, introducing adaptive learning rates to improve the optimization process. The key to its functionality lies in its ability to adapt the learning speed for each parameter individually. This adaptability is achieved by combining two important concepts: momentum methods and adaptive learning rates [1]

Adam momentum update.

The Adam optimization method calculates the average slope of the path (m_t) and the average steepness of the hills (v_t):

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t, \tag{1}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2. \tag{2}$$

where β_1 and β_2 are hyperparameters controlling exponential velocities,
g_t is the gradient of the loss function with respect to the model parameters at iteration t.

Bias correction.

There is a slight correction to make sure that these averages are fair, especially at the beginning.

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \tag{3}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t}, \tag{4}$$

where $\hat{m}_t$ represents the exponentially weighted average of past gradients,
$\hat{v}_t$ represents the exponentially weighted mean of past quadratic gradients,

Parameter update.

It then uses these averages to decide how large the steps should be for subsequent iterations. The model parameters are updated according to the equation below.

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t + \varepsilon}} \hat{m}_t, \tag{5}$$

where *α* is a learning rate,ε and is a small constant that prevents division by zero.

Let's consider the deeper meaning of these hyperparameters. Understanding and fine-tuning hyperparameters is critical to harnessing the full potential of the Adam optimizer in neural network projects. Properly tuning them ensures optimal convergence and performance for most machine learning or deep learning tasks.

- Learning step (α): controls the size of the step during optimization. A higher learning rate may result in faster convergence but may also result in missing the minimum. Requires careful adjustment.
- β_1 and β_2: exponential recession rates for past gradients and quadratic gradients, respectively. They affect the weight of recent and past information. Common default values: 0.9 for β_1 and 0.999 for β_2.
- Epsilon (ε): a small constant to prevent division by zero. As a rule, it has a small value, for example, 10^{-8}.

The speed of adaptive learning in Adam plays a crucial role in solving the problems faced by traditional optimization algorithms. Adam dynamically adjusts the step size of each parameter, ensuring a balanced and efficient convergence process. This adaptability mainly makes adaptive optimizers like Adam the best algorithms when dealing with sparse gradients, non-stationary targets, or different gradient sizes for different parameters.

Adam's performance may depend on the choice of learning speed. In some cases, an incorrect setting of the learning speed can lead to suboptimal convergence or method divergence. Thus, in particular, in the work [2] it was noted that depending on the size of the learning rate of the α, there may be different modes of learning, namely: non-additional training, the mode of satisfactory training, and overfitting with the transition to a chaotic mode of training. Logically, the interval of their existence in α depends on the number of iterations.

Adam supports additional state variables (moment estimates) for each parameter, which increases memory requirements. For large deep learning models or datasets, this can be a limitation, especially in environments with limited memory.

Unlike some simpler optimization algorithms, Adam does not have strong theoretical guarantees of convergence. Although in practice it often works well. The lack of clear theoretical guarantees can make it difficult to predict its behavior in specific scenarios.

In cases where the target function has a lot of noise, Adam may not work optimally. Noise-related gradients can affect the rate of adaptive learning and moment estimation, resulting in suboptimal convergence.

Adam's performance depends on the initial values of his moment estimates (m and v). In some cases, incorrect initialization can affect convergence.

The additional computations required to maintain and update moment estimates make Adam computationally more expensive compared to simpler optimizers, especially in scenarios with large models and datasets.

In some cases, Adam's impulse-like behavior can cause the minimum to be exceeded, especially in the large dimension parameter space.

A number of publications are devoted to the Adam optimization method. Here are some key publications devoted to this optimization method. In the paper [3], the authors first presented the Adam algorithm, which combines the advantages of the Momentum and RMSprop methods for adaptive adjustment of the learning rate during stochastic optimization. The author's article by Rahul Agarwal [4] provides a detailed overview of the Adam algorithm, its advantages, drawbacks, and practical aspects of use, which can be useful for machine learning practitioners. In the paper [5], the authors propose a modification of the Adam algorithm, known as AdamW, which improves the regularization of the model by separating the weight decay from parameter updates, which contributes to better generalization of the model. The paper [6] analyses the problem of large variation in adaptive learning rate in Adam-type algorithms at the initial stages of training and proposes a variant of RAdam, which fixes this issue and improves the stability and performance of optimization. The paper [7] presents the NosAdam algorithm, which gives more weight to the previous gradients in calculating the adaptive learning rate, which contributes to better convergence and stability of learning. The paper [8] investigates the choice of optimal learning rates for adaptive algorithms such as Adam and AMSGrad and their impact on the convergence and performance of deep neural networks.

In the paper [9], we present a new and complex framework for analyzing the convergence properties of Adam. This structure offers a universal approach to establishing Adam convergence. Adam has been shown to reach non-asymptotic limits of sample complexity similar to SGD.

The purpose of this work is to study maps of dynamic learning modes of a multilayer neural network using the Adam optimization method for the target learning error function.

The hyperparameters β_1 and β_2 were scanned within the interval [0; 0.999] using a uniform scanning step. The periodicity detection parameter q was chosen as q = 8 or q = 16, depending on the experiment. The neural network was trained using the mean squared error (MSE) loss function. Sigmoid activation was used in the output layer, while standard nonlinear activation functions were used in hidden layers. Weights were initialized randomly with a fixed random seed to ensure reproducibility. A regime was classified as chaotic if no periodicity up to order q was detected and convergence was absent within the observation window.

MATERIALS AND METHODS

The construction of dynamic mode maps was carried out in the coordinates of the hyperparameters at different values of the learning rate, the number of hidden layers, and the number of neurons in them and the number of iterations. Under these conditions, the learning error value of each neuron was analyzed in comparison with the others. The construction of a multilayer neural network was performed in the Python software environment. An array of printed digits represented as a set of zeros and ones with a dimension of 4×7 was used as the dataset. The sample contained 5 options for representing each digit. The sample for each digit contained variants of distortions of this digit. The amount of distortion was 5%. The construction of maps of dynamic modes was carried out according to the method described in the paper [10] and was as follows.

It is known that two-dimensional mappings can be specified by recurrence relations of the form:

$$\begin{aligned} x_{n+1} &= f(x_n, y_n), \\ y_{n+1} &= g(x_n, y_n). \end{aligned} \tag{6}$$

When building maps of dynamic modes, the function $f(x_n, y_n)$ was a function describing the learning error of a neuron in the output layer on the n-th neuron. The function $g(x_n, y_n)$ was a function describing the learning error of a neuron in the output layer on the $(n-1)$-th neuron.

The following algorithm was used to build maps of dynamic modes.

For some initial pair (x_0, y_0) at some pair of values (β_1, β_2), according to Eq. (6), a set of iterative values $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$.

A few q last points stand out. For example, 8 or 16. q is the number that will denote the maximum period that can be determined using this algorithm. The selected points form a set of $(x_{n-q+1}, y_{n-q+1}), (x_{n-q+2}, y_{n-q+2}), \ldots, (x_n, y_n)$.

Working with a set of points that has 2 coordinates is not very convenient in our case. To do this, you need to replace 2 coordinates with a single number. For example, replace with a square the modulus of the vector that corresponds to this coordinate. That is, instead of a point (x_i, y_i), the value $d_i = x_i^2 + y_i^2$ is used. The result is a set of $d = \{d_1, d_2, \ldots\}$. For further actions, it will be necessary that d_i displays a specific value (x_i, y_i) and does not allow duplicates. But this cannot be achieved if the points form a certain symmetrical figure, for example, a circle – the distances to the middle of the center of the coordinates of which will be the same. Therefore, it makes sense to introduce some asymmetry into the above operation. For example, like this: $d_i = (x_i + k_1)^2 + (y_i + k_2)^2$.

If the last, q element, coincides with $q-1$, then it is stated that the resulting set d has a period of 1, which means that with the parameters (β_1, β_2), the system has a limit point. until the period is detected. That is, where a situation is reached when none of the numbers of the set d will coincide with the last element. In this case, the presence of chaos is asserted. Although in fact it may turn out to be a period of higher order or the system has not yet moved to stable behavior, and we ourselves have the right to choose with what accuracy the behavior of the system is considered. In a situation where the period has not been detected, it is wise to write it as zero. The above method of determining the period can be called the method of simple comparison or the method of convergence of periodicities.

Next, it is necessary to save the obtained result. In pre-created 3 arrays, values are stored according to the following rule: in the first array – the value of β_1, in the second – the value of β_2, and, finally, in the third – the value of the obtained period.

All the previous points are worked out at other values of β_1, β_2. It is necessary to go through all possible combinations (β_1, β_2) from the interval, going over with a certain step.

The last stage is the display of the image. Points corresponding to different period values must be displayed using different colors. To do this, you can use check loops or logical operations.

When building dynamic mode maps, we selected only the first fourteen possible periodicities (red – 1, orange – 2, yellow – 3, green – 4, cyan – 5, blue – 6, violet – 7, navy (#000080) – 8, light shade of purple-blue (#9370DB) – 9, dark -orchid (#9932CC) – 10, medium light shade of purple (#DDA0DD) – 11, purple red (#C71585) – 12, midnight blue (#191970) – 13, purple (#DB7093) – 14, white (white) – all others). The black color indicates the absence of any periodicity.

RESULTS AND DISCUSSION

Fig. 1 shows maps of dynamic training modes of a three-layer neural network for printed digits with 28 neurons in each layer, training step at 10 iterations. Maps of dynamic neuronal learning modes are characterized mainly by the existence of the first periodicity (red color) in the entire spatial area of change. Since the first periodicity is associated with the largest gradient change, this indicates that under these conditions of neural network training, its underfitting can be traced. Only in the spatial region of changes (0.7–0.9999) and (0–0.2) there is an appearance of periodicity with a longer period (tripling and above), that is, a decrease in the gradient of weight correction, and therefore in the speed of learning. Therefore, under these conditions (during the training step and 10 iterations), the neural network is in underfitting mode.

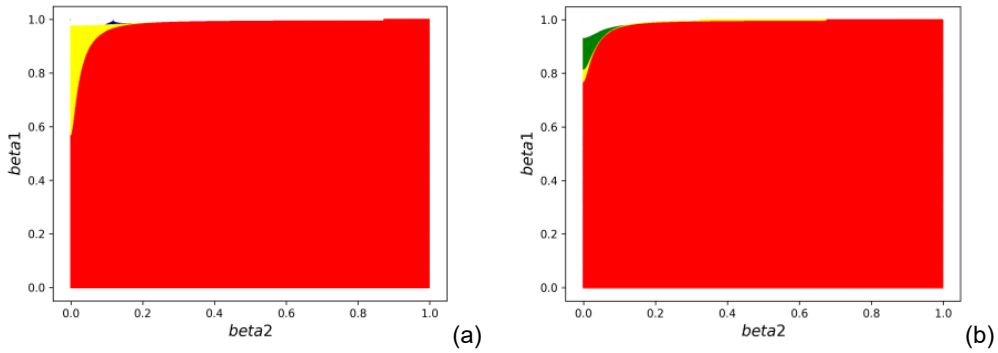

Fig. 1. Dynamic regime maps of a three-layer neural network trained on printed digits, with 28 neurons per layer, learning rate $\alpha = 0.001$, and 10 training iterations: (a) digit “0”; (b) digit “1”, illustrating the dependence of learning dynamics on the input pattern

Fig. 2 shows maps of dynamic learning modes of the neural network, provided that the learning rate (α) is equal to 0.1, with 10 iterations. With small values of (0–0.2) and (0–0.2), the existence of periodicity with a longer period can be traced.

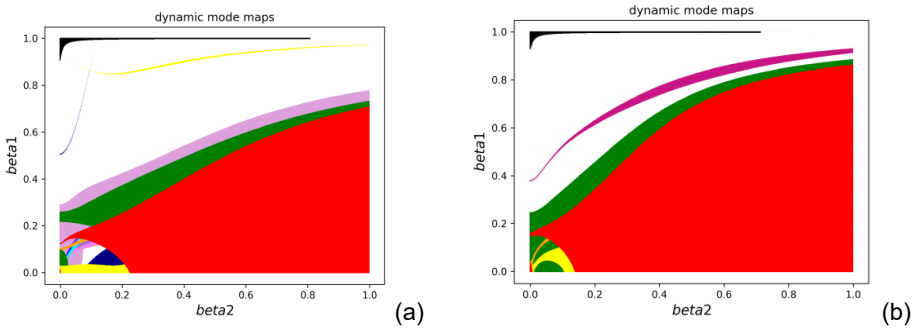

Fig. 2. Dynamic regime maps of a three-layer neural network trained on printed digits, with 28 neurons per layer, learning rate $\alpha = 0.1$, and 10 training iterations: (a) digit “0”; (b) digit “1”, illustrating the dependence of learning dynamics on the input pattern

The update of weights is more sensitive to gradients, and in this interval of changes in optimization parameters, an uneven learning process is observed. In this range of changes β_1 and β_2, the emergence and increase of the range of periodicity with the smallest period can be traced. That is, training takes place with a larger gradient of weight correction. With an increase in optimization parameters β_1 and β_2, there is an expansion of the spatial area of existence of periodicity with the shortest period. At values of β_1 (0.2–0.6) and β_2 (0.2–0.999), it (the periodicity with the smallest period) becomes decisive. That is, the learning process takes place at the same speed. Along with this, for some numbers ("0", "2", "4", "5", "6", "9"), there is a threefold (yellow) period, four (green) longer than the periodicity with the shortest period (red). At $\beta_1 > 0.9$ and $\beta_2 > 0.9$, according to the authors, there is a transition to periodicities with a period longer than the existing largest periodicity (white palette (white) - all other periodicities). Therefore, high values of β_1 and β_2 stimulate a smooth update of learning parameters with a transition to insignificant, large gradients of weight correction, or practically to their absence. That is, under these conditions $\beta_1 > 0.9$ and $\beta_2 > 0.9$ (white palette (white) - all other periodicities), there may be no process of correcting the weights. Based on the above, the following conclusion can be drawn. Provided that the control of inertia in the direction of the gradients is negligible ($\beta_1 < 0.4$) and the control of the "adaptive" learning rate is insufficient ($\beta_2 < 0.2$), it reacts quickly to gradient changes, resulting in unstable learning and oscillations. Under the same conditions, the speed adapts to new gradients, and this increases the risk of fluctuations (there is no change in the colors specified in the method, and therefore periodicity.

It is known [11] that in the Adam (Adaptive Moment Estimation) method, the parameters β_1 and β_2 control the smoothing of first- and second-order moments, respectively. They affect the speed of updating weights and the stability of learning. β_1 (usually 0.9) determines the smoothing of the first moment (mean gradient), expression (1.1)

A high β_1 (for example, 0.4–0.99) adds "inertia" to the update of the weights, which helps to avoid sudden changes in the direction of movement, as well as smoothes gradients, stabilizes learning.

Low β_1 results in less anti-aliasing effect, making weight updates more sensitive to current gradients, and making learning faster but less stable.

β_2 determines the smoothing of the second moment (the mean square of the gradients), the expression (1.2).

A high β_2 allows for smooth parameter updates, but can cause excessive anti-aliasing, which slows learning, as well as smoother refresh but slower adaptation of learning speed [12].

Low β_2 makes the model more sensitive to gradient changes, which can help when gradients change rapidly, as well as faster adaptation, but greater risk of instability.

Usually, the default is β_1 = 0.9, β_2 = 0.999 [11]. If the gradients change very quickly, you can reduce β_2. If the weights are very unstable, you can increase β_1.

So, based on **Fig. 2**, at $\beta_1 > 0.9$ and $\beta_2 > 0.9$, there is a smooth update of parameters, which is accompanied by a slowdown in learning or practically its absence. To identify the learning dynamics processes of a neural network under the condition $\beta_1 > 0.9$ and $\beta_2 > 0.9$, we will consider the influence of the number of iterations, the learning rate, the number of hidden layers and neurons in them on the learning process.

Influence of the number of iterations.

Fig. 3 shows the dependence of dynamic mode maps on the number of iterations. The obtained maps of dynamic modes of neuronal learning indicate that the process of difference in the learning of one neuron in relation to another is not very sensitive to the number of iterations.

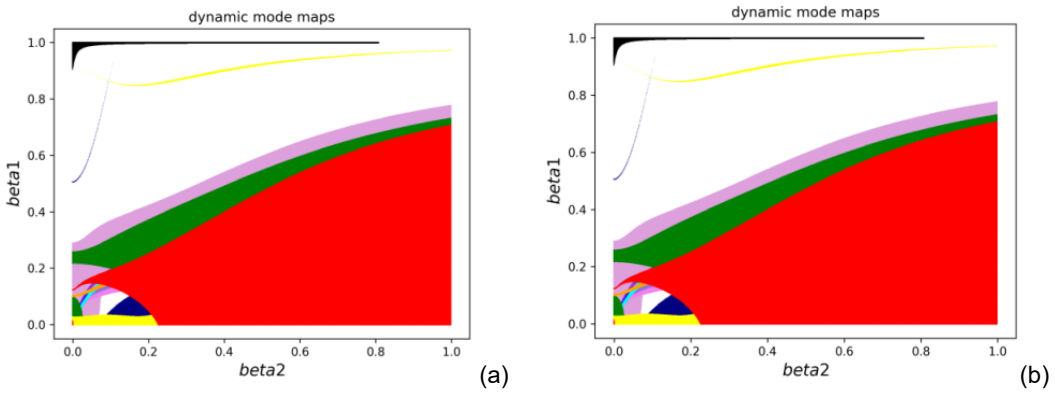

Fig. 3. Dynamic regime maps of a three-layer neural network trained on printed digits (digit "0"), with 28 neurons per layer and learning rate α = 0.1: (a) 3 training iterations; (b) 100 training iterations, illustrating the effect of iteration number on learning dynamics

Although it is known [13] that the number of iterations directly affects the role of the β_1 and β_2 parameters in the Adam method, especially because of their role in bias correction in the early stages of learning. Since the evaluation moments of m_t and v_t are calculated recursively, they are shifted towards zero in the first iterations. To compensate for this, bias correction is applied to expressions (1.3) and (1.4). This correction has a strong effect in the early stages of learning (small t). Since β_1 and β_2 are initially close to zero, the denominator is small, which amplifies updates in early iterations.

With a large number of iterations ($t \to \infty$), correction becomes almost unnecessary, since $1 - \beta_1^t \approx 1$ and $1 - \beta_2^t \approx 1$. On small iterations, low values β_1 make the weights update sensitive to gradient changes. High β_1 gives greater inertia, which stabilizes the

update. On large iterations, the effect of choosing β_1 is smoothed out because the middle gradient accumulates enough information.

If the training is short (a few iterations), it is worth reducing β_1 to respond faster to changes. If the training is long, high β_1 (0.9–0.99) improves stability.

On small iterations, high β_2 makes updating weights very careful, which can slow adaptation. On large iterations: v_t (RMS gradients) is already stabilized, and β_2 is affected less. If the learning is short (a few iterations), you can reduce β_2 it so that the adaptation of the learning speed is faster. If the training is long, high β_2 will help to avoid fluctuations in the learning speed.

Therefore, in the early iterations, the bias correction plays an important role by enhancing the effect of the gradients. In the later iterations, the effect of choosing β_1 and β_2 becomes less critical, and Adam works stably. If the training is short (a few iterations), it is possible to reduce β_1 and β_2 so that the network adapts faster. If the training is long, the standard values are β_1 = 0.9, β_2 = 0.9999, and usually work well [12, 13].

The difference between our results and the generally accepted statements about the influence of the number of iterations on the role of the parameters β_1 and β_2 in the Adam method may be related to the method for determining recurrence relationships, Eq. (1.6).

Impact of the learning rate.

The learning rate (α) in the Adam optimizer significantly affects the learning process of the neural network [3]. This manifests itself as follows.

The Large learning rate (α).

A quick update of the weights, which can accelerate convergence, while the network may "jump" the optimum and not stabilize. The error can fluctuate or even increase.

Small learning rate (α).

Learning is stable and smooth. Very slow convergence. Can get stuck at a local minimum.

The optimal value of the learning rate would be the balance between speed and stability. Normally, Adam works well at $\alpha = 0.001$ (recommended value for convolutional neural networks). A reduction in the learning rate can be used to improve convergence [3].

Choosing the right training step is critical for the neural network to train effectively.

According to the work [2], an increase in the learning rate (α) can lead to overfitting of the neural network, which ultimately leads to a chaotic learning mode of the neural network (for a three-layer neural network). **Fig. 4 a–c** shows maps of dynamic modes under the condition of $\alpha = 0.4, \alpha = 0.6, \alpha = 0.9$. According to **Fig. 4a**, an increase in the learning rate caused a decrease in the spatial region in the coordinates β_1 and β_2 the existence of periodicity. In particular, first of all, periodicity with the shortest period. Along with this, the appearance of new periodicities and the expansion of existing periodicities with an increased period take place. This indicates a change in the speed of learning in the direction of its decrease. In our opinion, this is due to an increase in the speed of weight correction and the rapid achievement of a local minimum.

Let's consider it in more detail. Under the condition of $\beta_1 < 0.1$ and $\beta_2 < 0.9$, there is practically no learning process. A similar situation is observed for $0.4 < \beta_1 < 0.99$ and $0 < \beta_2 < 0.99$. The origin of periodicity with different periods can be traced in the spatial region of changes in optimization parameters $0 < \beta_1 < 0.3$ and $0 < \beta_2 < 0.2$. A further increase in the β_2 parameter is an increase in the spatial regions of periodicity. In particular, this is well traced for the smallest periodicity, and for the periodicity of which

the period has increased by three, four, five, and six times (**Fig. 4a**). Ideally, the pattern of neural network learning when using an optimizer should be as follows.

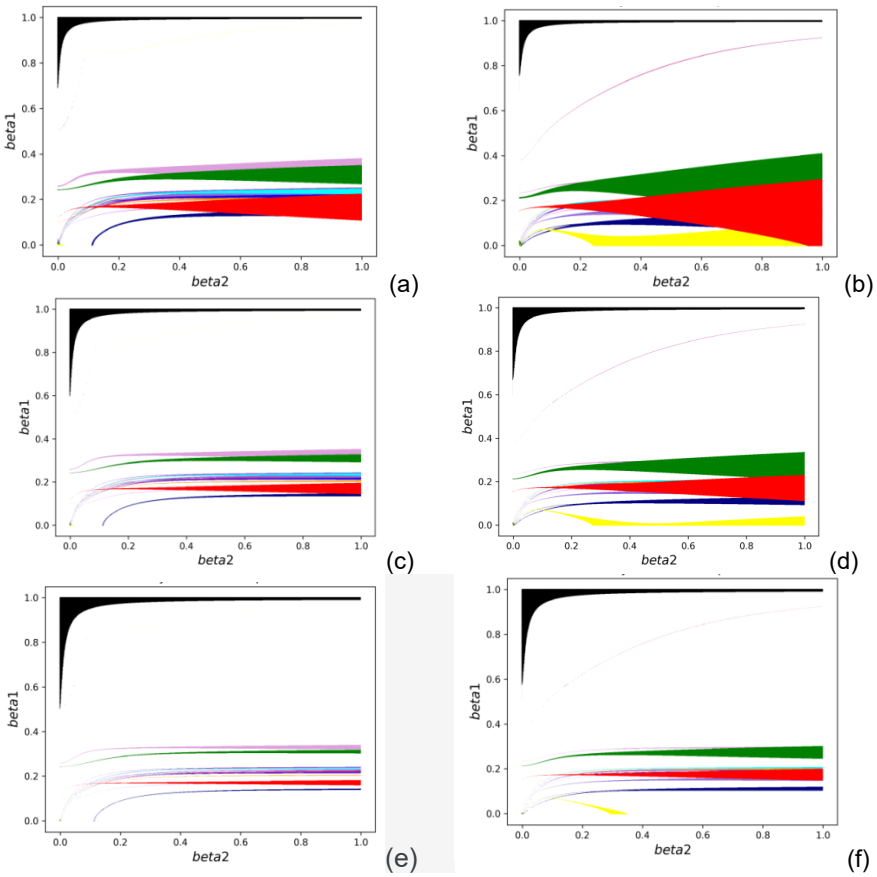

Fig. 4. Dynamic regime maps of a three-layer neural network trained on printed digits, with 28 neurons per layer and 10 training iterations: (a), (b) learning rate α = 0.4; (c), (d) α = 0.6; (e), (f) α = 0.9; for each pair, (a), (c), (e) correspond to digit "0" and (b), (d), (f) to digit "1", illustrating the effect of increasing learning rate on learning dynamics and the emergence of complex and chaotic regimes

At the beginning, as a result of large gradients, the highest learning speed should be traced, in our case, the correction Weights. As we approach the global minimum, the magnitude of the gradients decreases, so there will be an increase in periodicity. That is, ideally, we should observe the following sequence of color palettes with increasing values of optimization parameters β_1 and β_2: red – 1, orange – 2, yellow – 3, green – 4, cyan – 5, blue – 6, violet – 7, navy (#000080) – 8, light shade of purple-blue (#9370DB) – 9, dark – orchid (#9932CC) – 10, medium light shade of purple (#DDA0DD) – 11, purple red (#C71585) – 12, midnight blue (#191970) – 13, purple (#DB7093) – 14, white (white) – all others). That is, the first periodicity (red color) with the smallest period is described by the largest changes in the learning error gradient. Other, subsequent periodicities are characterized by a longer period multiple of the period of the first periodicity. In fact, according to **Fig. 4a**, at the initial stage, learning begins with not the biggest change in the target learning function of the neural network. That is, learning begins with periodicities, the period of which is three or six times longer than the period of the first periodicity. This indicates that learning does not begin with the highest learning speed, but three to six times less than the maximum possible. The sequence of periodicities with

an increase in the value of β_1 in the entire range of changes in the optimization parameter β_2 is also far from the ideal sequence of periodicities.

That is, for most digits, except for "1", "2", "3", "4", "8", the periodicity with a longer period is traced first, before that with a decreasing period. Therefore, there is a non-equilibrium learning process, which is accompanied by a different sequence of periodicities, i.e. uneven learning with different changes in speeds when changing parameters. In addition, it should be noted an increase in the spatial area of changes in the parameters β_1 and β_2, where periodicity is traced, the period of which is 14 times longer than the period of the first periodicity. In this area, chaotic fluctuations in the speed of neurons learning can also occur.

A further increase in the learning rate $\alpha = 0.6$ (**Fig. 4b**) is similar to that for **Fig. 4a**.

The sequence of periodicity with increasing magnitude β_1 in the entire range of changes in the optimization parameter β_2 is also far from its ideal sequence. This indicates a non-equilibrium learning process, which is accompanied by a different sequence of periodicities, i.e. uneven learning with different changes in speeds when optimizing parameters change. In addition, there is an increase in the spatial area of changes in parameters β_1 and β_2, and where periodicity is traced, the period of which is 14 times longer than the period of the first periodicity.

Fig. 4c shows maps of the dynamic learning modes of a neural network during a learning rate $\alpha = 0.9$. The resulting maps are dynamic, provided $\alpha = 0.9$, indicate the practical absence of the neural network learning process. The observed periodicities have an insignificant spatial area of existence and undergo their non-sequential change (with a decrease in the period). That is, the learning process is almost non-existent. It is known [2] that with a given value of the learning rate in multilayer neural networks, the existence of a chaotic learning mode can be traced, which is due to the overfitting of neurons. The optimizer's attempt to stabilize the learning process leads to the existence of a number of periodicities with a narrow area of their existence, which can be traced in **Fig. 4b** and **Fig. 4c**.

Additional studies of dynamic mode maps have been carried out, under these conditions, at different values ε (ε determines the existence of periodicities with a value of ε), testified to the absence of any periodicity in the spatial region, which is described by the existence of white. This state of the neural network can be described as a chaotic mode of the neural network learning process. This conclusion follows from the algorithm for building maps of dynamic modes using the method of convergence of periodicity. Namely, it is known that the chaotic regime is characterized by the fact that small changes in the initial conditions lead to radically different results. Under such conditions, periodicities occur that are not convergent, and the amplitude of such fluctuations can vary sharply. Therefore, in this method, a limit was set on the magnification of the learning error, and on the map of dynamic modes, this area is indicated in black. An area that is represented in white can be described as a region in which there are periodicities with a period greater than fourteen times the periodicity with the smallest period, zero period, or chaotic. The chaotic oscillations existing in this area (indicated in white) are amplitudes smaller in magnitude than in the specified black region. Therefore, in the white area of dynamic mode maps, a chaotic learning mode occurs with an amplitude less than the specified value in the black area.

Influence of the number of neurons

It is known [3–8] that the change in the number of neurons in the network affects the stability and efficiency of the neural network learning process.

If the number of neurons is small, Adam works stably, because there are fewer parameters and fewer gradient fluctuations. The model learns quickly. Possible

underfitting is insufficient expressiveness of the model. Adam updates the weights, but they are not enough for the model to summarize the data well.

You can use a larger learning rate (α), because the model is less sensitive to large updates. Gradients are more stable, so moments work efficiently [3].

If the number of neurons is large, the model can study complex patterns. The gradients become noisy, which can make it difficult for Adam to work. The model can be overtrained. Learning becomes slower due to more scale updates. Under these conditions, it is necessary to reduce the training step (α) to avoid sudden changes in large models. Adam adapts to gradient changes but can match more slowly due to the large size of the parameters.

So, when there are fewer neurons, Adam works more stably, and a larger learning rate can be used. More neurons – Adam can be unstable due to variable gradients, requiring a lower learning rate [3].

Fig. 5a–j presents dynamic regime maps for different numbers of neurons in the network. **Figures 5a–d** correspond to configurations with fewer neurons than the input dimensionality, whereas **Fig. 5e–j** illustrate cases with a larger number of neurons. For the configuration with 20 neurons per layer (i.e., fewer than the number of input features), the initial variation of the hyperparameters β_1 and $\beta_2$is associated with the emergence of first-order periodicity. As β_1 and β_2 increase, this periodicity expands and gradually transitions to higher-order periodic regimes. Notably, the observed periodicities may exceed the fundamental period by more than a factor of five. Thus, for small values of the optimization parameters ($\beta_1 < 0.2$ and $\beta_2 < 0.2$), the learning process is characterized by the dominance of low-order periodic behavior.

Learning is mostly characterized by an uneven learning speed. At low values of β_1 ($\beta_1 < 0.2$) and β_2 ($\beta_2 < 0.2$), rapid adaptation takes place, but with greater instability. Lower values of β_2 are accompanied by a smaller smoothing effect, and the update of weights is painfully sensitive to current gradients. A further increase in the optimization parameters stimulates the presence of the highest learning rate, the interval of existence of which mainly depends on the parameter β_1. That is, β_1 it affects the rate of renewal of weights and the stability of learning. At higher values of β_2, the intervals of the existence of periodicities are larger. This indicates that β_2 it causes excessive smoothing, and this results in homogeneous learning. This, in particular, is indicated by the sequence of existing periodicities. That is, according to the first periodicity, the presence of periodicity is observed, a period that significantly exceeds (five or more times) the period of the first periodicity. A further increase in β_1 and β_2 causes the disappearance of periodicity, which indicates the transition of the neural network to the overfitting mode or to the chaotic mode. Therefore, a sharp transition to the periodicity with the longest period indicates the presence of a smoothing process, which provokes a decrease in the learning speed and expansion of the spatial areas of the periodicity of periodicity on the map of dynamic modes.

A further decrease in the number of neurons (up to 18, **Fig. 5c-d**) is accompanied by the separation of the spatial region of changes β_1 ($\beta_1 < 0.4$) and β_2 ($\beta_2 < 0.4$), where the existence of a significant number of periodicities, which indicates the existence of a number of learning speeds. Further increase in the value of parameters β_1 and β_2 is accompanied by the expansion of the spatial regions of the first, second, and third periodicity. This indicates a gradual transition to a decrease in the value of the learning speed (periodicity determined by red, yellow, and orange). The next increase in the value of β_1 and β_2 contributes to the appearance of periodicity with an even longer period, which indicates a further decrease in the learning rate. In addition, there is a transition to the absence of any periodicity (areas of white color), which in some places are separated by periodicities with a long period. This behavior of dynamic mode maps at large values

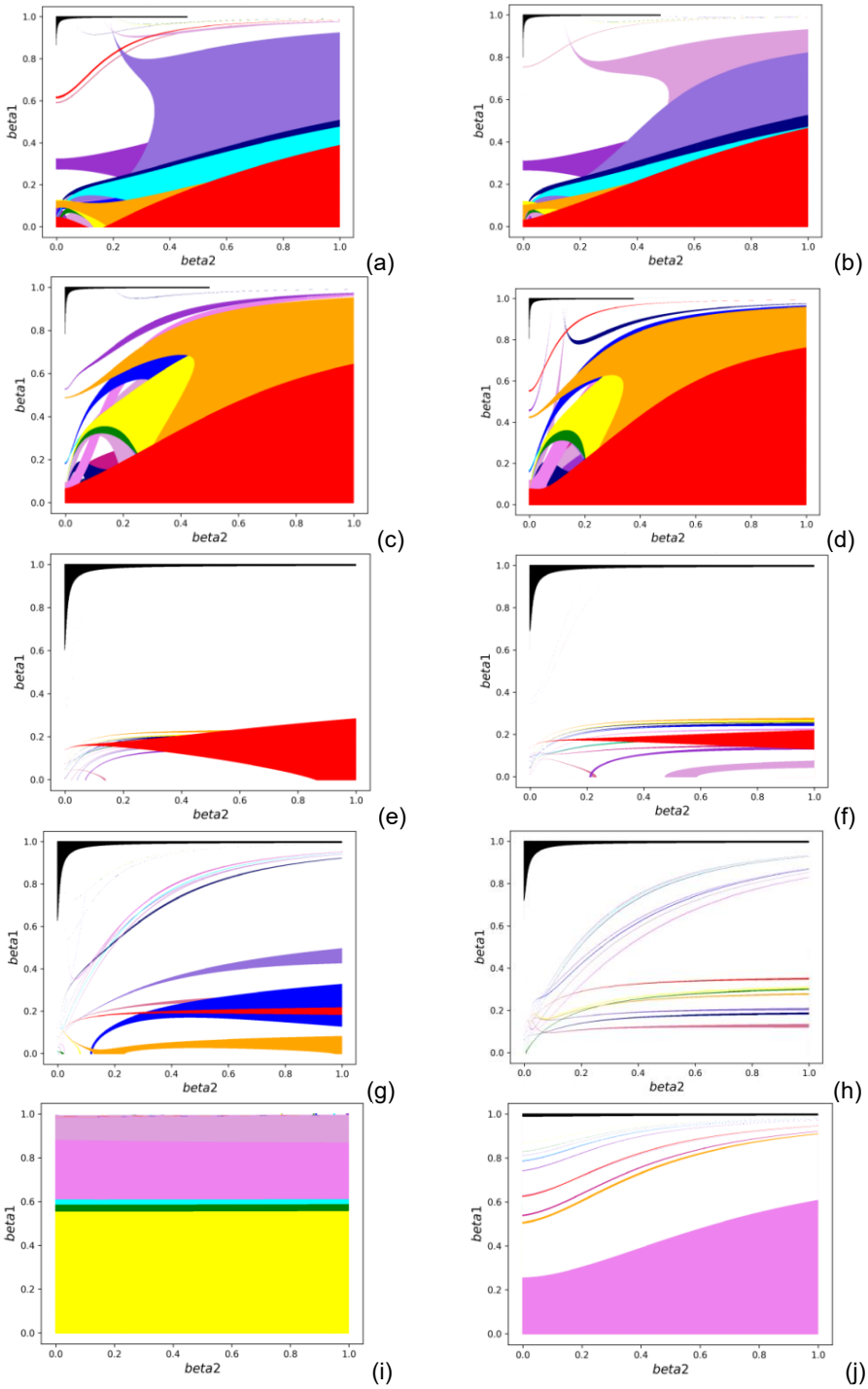

Fig. 5. Dynamic regime maps of a three-layer neural network trained on printed digits, evaluated over 10 training iterations: (a), (b) 20 neurons per layer, learning rate $\alpha = 0.1$; (c), (d) 18 neurons per layer, $\alpha = 0.1$; (e), (f) 50 neurons per layer, $\alpha = 0.4$; (g), (h) 75 neurons per layer, $\alpha = 0.4$; (i), (j) 100 neurons per layer, $\alpha = 0.4$; for each pair, the left panel corresponds to digit “0” and the right panel to digit “1”, illustrating the effect of network width on learning dynamics

of β_1 and β_2 indicates a significant decrease in the learning rate or its absence. That is, at large values of β_1 and β_2 reaches a global minimum, and neural network training can go into chaotic learning mode. Wedging of single periodicities with a long period may indicate the presence of a learning process confirming the existence of transparency windows on the branching diagram [2].

A slight further decrease in the number of neurons (up to 15 neurons) is accompanied by a lack of periodicity on the maps of dynamic learning modes of the neural network. This indicates the absence of a neuronal learning process. Therefore, a decrease in the number of neurons in comparison with the number of input variables leads to an improvement in the learning process in the early stages, making it more monotonous in terms of learning speed.

Let's consider how the process of increasing the number of neurons affects the dynamics of the learning speed. **Fig. 5e–j** shows maps of the dynamic modes of training of the neural network with an increase in the number of neurons in each layer (50, 75, 100 neurons). Choosing a learning rate equal α = 0.4 is due to the fact that, as shown in the paper [2], this training step corresponds to the optimal learning error of a three-layer neural network, with 28 neurons in each layer. An increase in the number of neurons is accompanied by a narrowing of the spatial areas of the existence of periodicities, the appearance of periodicities with longer periods. With 50 neurons (**Fig. 5e-f**), there is a sharp transition to periodicity with longer periods and to the mode of absence of the learning process. That is, there is a sharp decrease in the speed of learning and the transition to the mode of its absence. An increase in the number of neurons to 75 (**Fig. 5g-h**) leads to the separation of spatial areas of the existence of periodicity by areas of their absence, which indicates the appearance of a heterogeneous process of neuronal learning A further increase in the number of neurons (100 neurons, **Fig. 5i-j**) is accompanied by an increase in the heterogeneous process of neuronal learning, and possibly the appearance of a chaotic learning mode. This, in particular, is indicated by the results of work [2] on the influence of the number of neurons on the learning error of a three-layer neural network.

Consider how the number of hidden layers affects the performance of the Adam optimizer. **Fig. 6a–d** shows the influence of the number of hidden layers and the dynamics of the learning speed of the neural network.

An increase in the number of layers of a neural network from three to five is accompanied by a spatial expansion of the existence of periodicity with a shorter period, which indicates the existence of a slow learning process, compared to a three-layer neural network. At the same time, there is a decrease in the moment parameter ($\beta_1 < 0.4$), at which there is a more significant change in the learning speed. At $\beta_1 > 0.4$, the learning of the neural network goes into the mode of practically its absence. Wedging of single periodicities with a long period (**Fig. 6a-b**) may indicate the presence of a learning process with a low learning speed, which indicates the existence of a heterogeneous learning process. An increase in hidden layers from five to seven leads to the disappearance of almost periodicity with a short period (areas of red, orange, and yellow), and the appearance of periodicity with a longer period. That is, an increase in the number of layers of the neural network at this stage leads to a decrease in the learning speed of the neural network, and an increase in the heterogeneity of the learning process not only at $\beta_1 > 0.4$. Therefore, the learning process becomes more heterogeneous. This, according to the authors, leads to a decrease in learning speed. That is, the Adam optimization method becomes slower due to the fact that the gradients become noisier.

It is known [2] that the optimizer Adam adaptively changes the learning rate for each network parameter, taking into account the average and variance of the gradients.

Increasing the number of hidden layers affects the dynamics of gradients and, accordingly; the behavior of Adam.

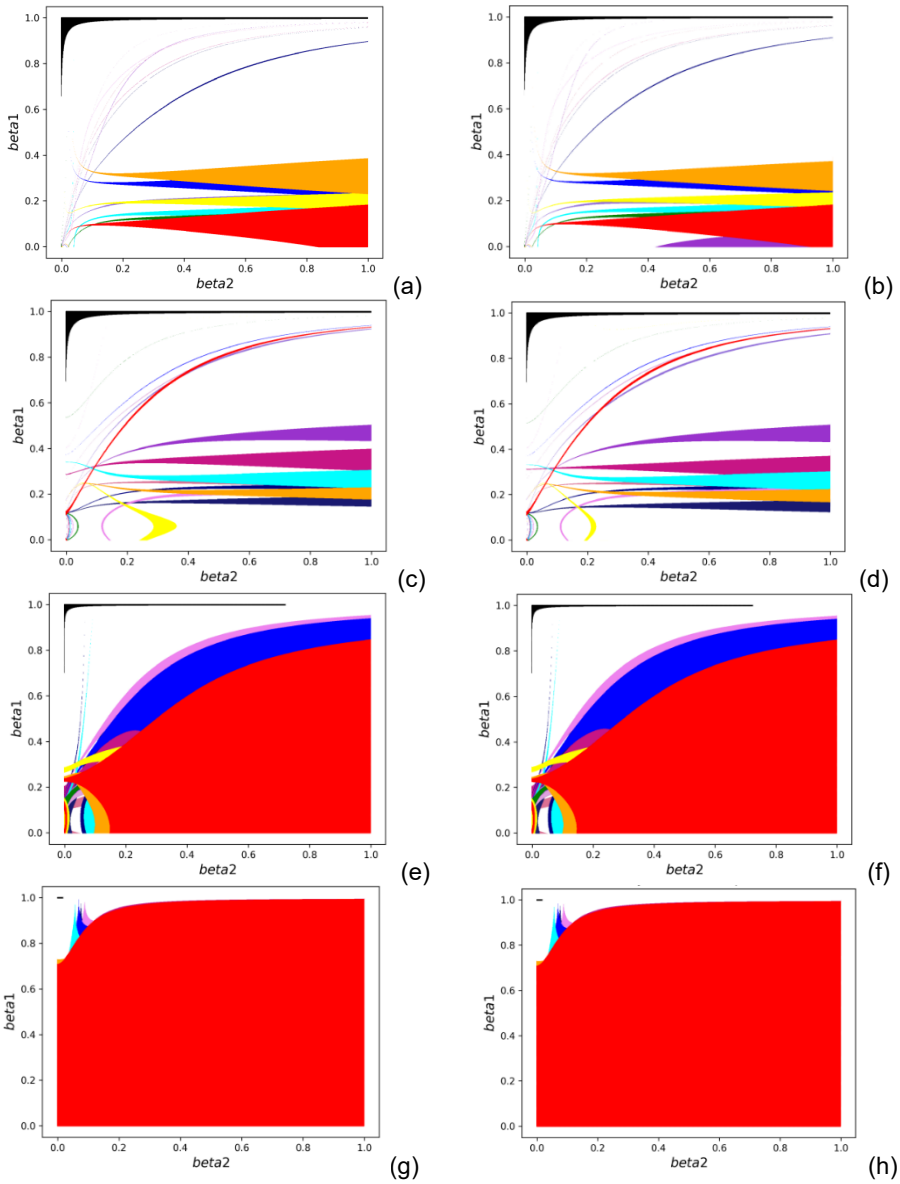

Fig. 6. Dynamic regime maps of a multilayer neural network trained on printed digits, with 28 neurons per layer, learning rate α = 0.4, and 10 training iterations: (a), (b) five-layer network; (c), (d) seven-layer network; (e), (f) nine-layer network; (g), (h) twelve-layer network; for each pair, the left panel corresponds to digit “0” and the right panel to digit “1”, illustrating the effect of network depth on learning dynamics

Few hidden layers (fine mesh). A simpler network is easier to optimize. Gradients change predictably, Adam controls them well.

Lots of hidden layers (deep web). Better at detecting complex patterns in data. May suffer from decay or explosion of gradients. Increasing the volume of calculations, Adam may run more slowly. Gradients can become noisy, making it difficult to learn.

Shown in **Fig. 6e–h** are maps of dynamic modes of the nine- and twelve-layer non-grid. Their maps of dynamic modes are radically different from maps of dynamic modes

with fewer layers (**Fig. 6a–d**). This difference lies in the presence of periodicity with the shortest period, which indicates an increase in the speed of learning. Taking into account the results shown in Fig. 6 g, h, according to the authors, this is due to the processes of undertraining of the neural network with a small number of iterations. Further studies with more hidden layers show a slowdown in the learning process.

So, the paper considers maps of dynamic learning modes of the neural network depending on the number of iterations, the learning rate, the number of neurons, and the number of neural network layers. The construction of maps of dynamic modes was carried out in the coordinates of hyperparameter optimization β_1 and β_2, and since these parameters, according to the work [2], determine the learning speed of the neural network. In the role of recurrent correlations was the learning error of the neuron and its neighbor. This is the peculiarity of the task.

The relationship was intended to determine the heterogeneity of the learning of one neuron in relation to another. Considering the influence of iterations (3, 5, 10, 100 iterations) on the dynamics of neural network learning, we did not establish any features of neuron training in relation to each other. As for the impact of the learning rate on the learning process, depending on its magnitude, there are modes of underfitting, satisfactory training (when there is no overfitting of the neurons) and the mode of overfitting, which degenerates into a chaotic one.

Let us now consider these learning modes in the coordinates of the hyperparameters of optimization β_1 and β_2. In the mode of underfitting (in particular, when $\alpha = 0.001$) in the range $0 < \beta_1 < 0.99$ and $0 < \beta_2 < 0.2$, a monotonic decrease in the learning rate is observed with an increase in the value of β_1. With higher values of β_2, $\beta_2 > 0.2$, the appearance of the underfitting process is traced in 10 iterations. It is known that a high value β_2 leads to a significant smoothing of correction gradients, which leads to a slowdown in the learning process. In the mode of satisfactory learning (in particular at $\alpha = 0.1$ and $\alpha = 0.4$), the neural network achieves a global minimum faster and transitions to a mode of practically no learning process. In the shift interval $0 < \beta_1 < 0.4$ and $0 < \beta_2 < 0.2$ traces the existence of abrupt and heterogeneous changes in the learning rate. An increase in the magnitude of the second moment (β_2) is accompanied by a smoothing of gradients, which leads to a monotonous decrease in the learning rate and the achievement of a global minimum. Under these conditions, if there is a heterogeneity of learning, it manifests itself in a consistent decrease in the speed of learning.

Switching to overfitting mode (In particular, when $= 0.6$, $\alpha = 0.9$), then at the coordinates β_1 and β_2 it manifests itself in heterogeneous learning, which is described both in a non-monotonic change in the learning rate and the existence of intervals where it is practically absent. It should be especially noted that at small values of β_1 in a satisfactory learning mode ($0.1 < \alpha < 0.5$), the appearance of a heterogeneous learning process, or practically no learning, can be traced. This is due to the magnitude β_1 and β_2 are much smaller than one. Since the evaluation of moments m_t and v_t are calculated recursively, they are shifted towards zero in the first iterations. To compensate for this, displacement correction (expressions (1.3) and (1.4)). This correction has a strong effect on early stages of training (small t), since by default the value β_1 and β_2, in Adam method, respectively, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. In our case, the parameters β_1 and β_2 are variable quantities and vary in the range 0, up to 0.999. That is, at small values, especially β_1, it does not occur Strengthening Training Updates. An increase in value β_2, with lower values of the learning rate, in the mode of satisfactory learning improves the situation somewhat, contributing to the learning process.

A change in the number of neurons in a multilayer neural network does not unambiguously affect the learning process when they increase compared to their decrease relative to the number of input values. On maps of dynamic modes in coordinates β_1 and β_2, this manifests itself as follows. If the number of neurons is less than the number of input values, Adam works stably. This is most likely due to fewer parameters and a lower value of gradient oscillations. The model learns quickly. The gradients are more stable, so the moments m_t and v_t work efficiently. The change in the learning rate in the coordinates β_1 and β_2 becomes more monotonous as the number of neurons decreases. In the interval of changes $0.1 < \beta_1 < 0.4$ and $0 < \beta_2 < 0.3$, the existence of sharp and heterogeneous changes in the learning speed is traced. An increase in the magnitude of the smoothing of the second moment (β_2) contributes to the smoothing of gradients, which leads to a monotonous decrease in the learning speed and the achievement of a global minimum. The model is less sensitive to large updates. Provided that the number of neurons is greater than the number of input variables, it contributes to the complication of the work of Adam. The constructed maps of dynamic modes testify to the overfitting of the neural network (50 neurons), with the transition to the chaotic mode of learning the neural network (75, 100 neurons). This is due to the noisiness of the gradients.

An increase in the number of layers is accompanied by an increase in the depth of the neural network, and therefore an increase in the complexity of its optimization. Built maps of dynamic modes in coordinates β_1 and β_2, indicate that with an increase in the number of hidden layers, at the initial stages (3÷7 layers), gradients become noisy, which complicates the learning process. Under these conditions, a heterogeneous learning process begins to be traced with the emergence of a chaotic learning mode. A further increase in the number of layers leads to the normalization of the learning process with the advent of the underfitting regime. The subsequent increase in the layers of the neural network contributes to the appearance of a satisfactory learning mode first, and then, with the transition to an overfitting mode, which turns into a chaotic one. At the same time, both attenuation modes and gradient explosions can be traced on dynamic mode maps.

The maps offer a compact diagnostic of Adam's stability across architecture and hyperparameters. Large β_1 and β_2 promote smooth updates but can stall progress; small β_1 and β_2 adapt rapidly but risk oscillations. Increasing network width/depth amplifies gradient noise, shrinking stable regions. These findings align with reports on optimizer sensitivity and the importance of tuning α and decay rates. Practically, the maps can guide the selection of (β_1, β_2, α) for a given budget of iterations and model size.

The obtained results should be considered as a qualitative framework for studying the nonlinear dynamics of the Adam optimizer rather than as a direct benchmark evaluation of modern datasets. Extension of the proposed approach to larger-scale datasets and architectures represents an important direction for future research.

CONCLUSION

Therefore, training a multilayer neural network with the Adam optimization method is sensitive to both the learning rate, the number of neurons in the layer, and the number of layers of the neural network. Changing these parameters leads to the emergence of several neural network training modes. In particular, to the mode of undertraining, satisfactory training, and the mode of overfitting, which can turn into a chaotic mode.

Recurrence-based dynamic mode maps in (β_1, β_2) reveal distinct Adam learning regimes and their dependence on α, iterations, width, and depth. This approach helps

identify stable operating regions and anticipate chaotic behavior in multilayer neural networks.

The source code is available in the GitHub repository at the following link: incom2025**neuro_map** (https://github.com/incom2025/neuro_map).

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [S.S., L.P., V.B.]; formal analysis, [S.S., I.K., Y.S.,].; investigation, [I.K.]; resources, [I.K.]; writing – original draft preparation, [S.S., I.K.]; writing – review and editing, [I.K., Y.S.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Kingma, D.P., Ba, J. Adam: A Method for Stochastic Optimisation. arXiv:1412.6980 (2014). https://doi.org/10.48550/arXiv.1412.6980

[2] Sveleba, S., Katerynchuk, I., Kuno, I., Karpa, I., Semotyuk, O., Shmyhelsky, Y., Sveleba, N., Kunyo, V. Chaotic states of a multilayer neural network. *Electronics and Information Technologies*. 2021;16:20–35. https://doi.org/10.30970/eli.16.3

[3] Loshchilov, I., Hutter, F. Decoupled Weight Decay Regularization. *arXiv:1711.05101* (2017). https://doi.org/10.48550/arXiv.1711.05101

[4] Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., Han, J. On the Variance of the Adaptive Learning Rate and Beyond. *arXiv:1908.03265* (2019). https://doi.org/10.48550/arXiv.1908.03265

[5] Huang, H., Wang, C., Dong, B. Nostalgic Adam: Weighting more of the past gradients when designing the adaptive learning rate. *IJCAI* (2019) 2532–2539. https://doi.org/10.24963/ijcai.2019/355

[6] Iiduka, H. Appropriate Learning Rates of Adaptive Learning Rate Optimisation Algorithms for Training Deep Neural Networks. *arXiv:2002.09647* (2020). https://doi.org/10.48550/arXiv.2002.09647

[7] Reddi, S.J., Kale, S., Kumar, S. On the Convergence of Adam and Beyond. In: *Proceedings of the International Conference on Learning Representations* (2018). https://doi.org/10.48550/arXiv.1904.09237

[8] Jin, R., Li, X., Yu, Y., Wang, B. A comprehensive framework for analyzing the convergence of Adam: bridging the gap with SGD. *arXiv:2410.04458* (2025). https://doi.org/10.48550/arXiv.2410.04458

[9] Choi, D., Shallue, C.J., Nado, Z., Lee, J., Maddison, C.J., Dahl, G.E. On Empirical Comparisons of Optimizers for Deep Learning. *arXiv:1910.05446* (2020). https://doi.org/10.48550/arXiv.1910.05446

[10] Sivaprasad, P.T., Mai, F., Vogels, T., Jaggi, M., Fleuret, F. Optimiser Benchmarking Needs to Account for Hyperparameter Tuning. *arXiv:1910.11758* (2020). https://doi.org/10.48550/arXiv.1910.11758

[11] Zhou, Z., Zhang, Q., Lu, G., Wang, H., Zhang, W., Yu, Y. AdaShift: Decorrelation and Convergence of Adaptive Learning Rate Methods. *arXiv:1810.00143* (2019). https://doi.org/10.48550/arXiv.1810.00143

[12] Sveleba, S., Katerynchuk, I., Kuno, I., Shmyhelsky, Y., Gut, S. Programs for calculating the map of dynamic modes using a system with chaotic modes. *Electronics and Information Technologies*. 2025; 30: 137-154. https://doi.org/10.30970/eli.30.11

БАГАТОПАРАМЕТРИЧНІ ДИНАМІЧНІ КАРТИ НАВЧАННЯ НЕЙРОННИХ МЕРЕЖ, НАВЧЕНИХ ЗА ДОПОМОГОЮ ОПТИМІЗАТОРА ADAM

Сергій Свелеба[1]*, Іван Катеринчук[1], Ярослав Шмигельський[1], Люціян Пельц[2], Володимир Бригілевич[2], Назар Свелеба[3]
[1]Львівський національний університет імені Івана Франка
вул. Ген. Тарнавського, 107, 79017 Львів, Україна
[2]Державна вища школа технологій та економіки в Ярославі
вул. Чарнецького 16, 37-500 Ярослав, Польща.
[3]Приватний вищий навчальний заклад «Європейський університет»,
бульвар академіка Вернадського 16 В, Київ, Україна

АНОТАЦІЯ

Вступ. Розуміння того, як гіперпараметри оптимізатора Adam формують динаміку навчання багатошарових нейронних мереж, залишається відкритою проблемою, особливо в режимах, де навчання переходить від стабільної до хаотичної поведінки. Попередні дослідження показували, що швидкість навчання та складність мережі можуть зумовлювати недонавчання, задовільне навчання, перенавчання або навіть хаотичні стани; однак систематичної візуалізації цих режимів у просторі параметрів β_1–β_2 досі бракує.

Матеріали та методи. Побудовано карти динамічних режимів для багатошарової нейронної мережі, навченої на двійково закодованих зразках цифр. Динаміку навчання аналізували за допомогою рекурентного двовимірного відображення, що базується на похибці нейрона та його сусіда. Періодичності у фінальних q ітераціях визначали методом збіжності періодичностей, що дало змогу класифікувати стани навчання – від стабільних фіксованих точок до циклів високого порядку та хаосу. Карти генерували для різних значень швидкості навчання α, кількості ітерацій, числа нейронів у шарі та глибини мережі, при цьому β_1 і β_2 сканували в інтервалі [0; 0.999].

Результати та обговорення. Дослідження показало, що поведінка Adam є сильно чутливою до спільних значень β_1 і β_2. Малі швидкості навчання ($\alpha = 0.001$) переважно приводять до періодичності першого порядку, що вказує на недонавчання. Помірні швидкості ($\alpha = 0.1 - 0.4$) формують структуровані переходи від швидкого навчання до повільніших режимів із зростанням β_2, тоді як надмірно великі значення α (≥ 0.6) породжують фрагментовані, немонотонні структури, характерні для перенавчання та хаотичної динаміки. Збільшення кількості нейронів або шарів посилює шум у градієнтах, звужуючи області стабільності та розширюючи хаотичні зони. Навпаки, зменшення числа нейронів нижче розмірності вхідного простору приводить до більш однорідних траєкторій навчання зі згладженими переходами періодичності.

Висновки. Картографування динамічних режимів у просторі β_1–β_2 є потужним діагностичним інструментом для визначення стабільних, неефективних або хаотичних режимів навчання в нейронних мережах, що навчаються за допомогою оптимізатора Adam. Результати підтверджують, що Adam є високочутливим до

взаємодії гіперпараметрів та архітектури мережі, і що некоректні їх комбінації можуть спричинити хаотичне навчання навіть за помірних швидкостей навчання. Запропонована методологія формує нову основу для аналізу динаміки оптимізаторів та оптимального вибору гіперпараметрів у моделях глибинного навчання.

Ключові слова: оптимізатор Adam; динаміка навчання; карти періодичності; налаштування гіперпараметрів; хаотичне навчання; багатошарові нейронні мережі.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
05 March, 2026	15 May, 2026	18 May, 2026	29 June 2026

Electronics and Information Technologies, 2026, 34, 51–64
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.89

SEMANTIC TEXT SIMILARITY ESTIMATION ACROSS DIVERSE LARGE LANGUAGE MODELS

Vitalii Parubochyi* , ***Oleksandr Chmykhalo*** ,
Oleh Husak , ***Roman Shuvar***
System Design Department,
Ivan Franko National University of Lviv,
50 Drahomanova Str., Lviv, 79005, Ukraine

Parubochyi, V., Chmykhalo, O., Husak, O., & Shuvar, R. (2026). Semantic Text Similarity Estimation Across Diverse Large Language Models. *Electronics and Information Technologies*, 34, 51–64. https://doi.org/10.30970/eli.34.3

ABSTRACT

Background. Semantic Text Similarity (STS) estimation is one of the useful proxy-based approaches for analyzing semantic differences between responses generated by various large language models (LLMs) used in AI chats and AI agents. Various studies in recent years have attempted to estimate STS for AI-generated responses, but they typically focus on ground-truth estimation by comparing AI-generated responses with predefined answers that, in most cases, differ structurally from the generated texts. However, cross-model STS estimation can provide valuable insights into model similarity, independent of how well they align with the predefined answers.

Materials and Methods. In this paper, we estimate both types of STS, but primarily focus on the STS between responses generated by several selected LLMs from different providers. We build our estimation framework using a subset of Frequently Asked Questions (FAQs) in English from the MFAQ dataset, as it offers a wide range of topics and a clear, easy-to-use structure.

Results and Discussion. The results of our experiments are organized into two parts. In the first part, we estimate ground-truth STS scores using multiple metrics (Cosine Similarity, BERTScore, and Latent Semantic Analysis (LSA)) to analyze how closely the generated answers from each model match the ground-truth answer in the MFAQ dataset. In the second part, we estimated cross-model STS scores using the same metrics to analyze how closely models align with each other. Additionally, we measure average request latency and use publicly available information on model context size and input and output prices to explore possible relationships between these parameters and STS-based quality indicators.

Conclusion. Our results indicate relatively high semantic similarity between models from the same provider in the selected experimental setup and demonstrate that high ground-truth STS scores do not necessarily imply high inter-model semantic similarity within the selected benchmark and metrics, as seen in the analysis of cross-model STS scores. Additionally, we show that not all metrics are equally suitable for cross-model STS estimation, and that metrics like Cosine Similarity or LSA scores can provide better sensitivity than BERTScore or similar metrics.

Keywords: semantic text similarity (STS), large language model (LLM), MFAQ dataset, cosine similarity, BERTS core, latent semantic analysis (LSA)

INTRODUCTION

Extremely rapid growth and evolution of large language models (LLMs) in the context of AI chats and AI agents make the problem of selecting an appropriate LLM for everyday tasks increasingly important. In most cases, different research and surveys compare cost, response speed (or request latency, depending on how you look at it), and answer correctness to address this question. When cost and response speed can be estimated objectively, the estimation of answer correctness is usually less reliable. It strongly depends on the chosen dataset and the additional settings provided to the LLM, often in the form of an additional prompt. Additionally, the criteria for correctness can vary, as well as the application area.

Much research focuses on ground-truth estimation [1-5], comparing model responses with predetermined answers that are usually not generated and structurally don't correspond to LLM responses. In this case, the correctness of the responses is usually measured by text similarity between the ground truth and generated responses. In other cases, the correctness can be measured as the consistency [3-4], coherence [3], or comprehensiveness [3] of the LLM-generated responses.

When we look at the application areas, most research focuses on specific domains, such as medicine [1, 5], education [3], market research [6-8], etc. In contrast, only a few focus on general knowledge that covers a wide range of topics.

In our research, we aim to examine LLM response estimation more broadly and focus on general knowledge, including information from fields such as medicine, retail, and technology. Therefore, we decided to use STS as a proxy indicator of semantic proximity to reference answers, while recognizing that STS does not directly measure factual correctness, completeness, or safety. That reduces sensitivity to structural differences between the reference and generated responses, but also to estimate cross-model STS, which enables us to indirectly analyze similarities and differences between LLMs. For the estimation dataset, we chose the Multilingual FAQ Dataset (MFAQ) [9], which contains more than 6M question-answer pairs across 21 languages. Though we used only question-answer pairs in English for our estimation framework and restricted the MFAQ subset to 1000 pairs from the validation dataset due to constraints on available LLM resources, we still estimated STS on a sample of question-answer pairs drawn from 883 web domains.

The most complex part of our estimation framework was selecting LLMs. We aimed not only to choose models from different LLM providers that, in theory, should yield different results, but also to estimate the different model classes and service tiers, such as Fast, Thinking and Pro, that model providers typically use to accommodate different LLM choices. In addition, we should have accounted for constraints across different LLMs when choosing models to fully estimate our test set. Therefore, we analyzed the available LLM providers and the models they offer, including not only the latest models, which are usually more expensive and have stricter constraints, but also earlier model generations (see **Table 1**). Based on it and available LLM resources, we chose two LLM providers and their corresponding models for the different model classes and service tiers:

- OpenAI: GPT-5-mini, GPT-4.1;
- Anthropic: Claude Haiku 4.5, Claude Sonnet 4.6, Claude Opus 4.6.

Despite the growing number of benchmarks for LLMs, there remains a methodological gap between evaluating reference-based responses and benchmarking model behavior. Existing approaches typically evaluate how close a generated response is to a predefined reference, but they do not sufficiently explain how semantically similar responses are between different models, nor do they support model selection given latency and cost constraints. This poses a scientific challenge: to demonstrate an integrated framework for assessing both response adequacy and semantic consistency between models. Therefore, the scientific objective of this study is to develop and

Table 1. **Overview of LLM providers and their models by operational mode**

LLM Provider	Fast / Small	Thinking / Medium	Pro / Large
2026 – Present			
OpenAI (GPT-5 Era)	GPT-5.4-mini	GPT-5.4	GPT-5.4-pro
Google	Gemini 3 Flash	Gemini 3 Thinking	Gemini 3.1 Pro
Anthropic	Claude Haiku 4.5	Claude Sonnet 4.6	Claude Opus 4.6
Meta AI	Llama 4 Scout	Llama 4 Maverick	Llama 4 Behemoth
DeepSeek	DeepSeek-V3.2	DeepSeek-R2	DeepSeek-V3.2-Speciale
Mistral AI	Mistral Small 4	Mistral Medium 3.1	Mistral Large 3
xAI	Grok-4.1 Fast	Grok-4 Heavy	Grok-4 o3
2023 – 2025			
OpenAI (GPT-4 Era)	GPT-4o mini	GPT-4o / Turbo	OpenAI o1-preview
Google	Gemini 1.5 Flash	Gemini 1.5 Pro	Gemini 1.0 Ultra
Anthropic	Claude 3 Haiku	Claude 3.5 Sonnet	Claude 3 Opus
Meta AI	Llama 3 8B	N/A	Llama 3 70B
Mistral AI	Mistral 8x7B	N/A	Mistral Large
DeepSeek	DeepSeek-R1-Lite	N/A	DeepSeek-V2

experimentally validate an STS-based framework that combines reference-based comparison, inter-model comparison, and operational indicators for a set of LLMs evaluated on a general-domain FAQ dataset.

In the next sections, we provide a detailed overview of related works, analyze the advantages of semantic text similarity, and discuss the metrics that can be used to estimate it. In the last two sections, we provide a detailed description of our estimation framework, how we obtained our test set from the MFAQ subset, analyzed the results, and draw insights from them.

RELATED WORK

Since the release of ChatGPT in November 2022 [10], the massive adoption of LLMs in the form of AI chats and agents has affected many people's daily lives in many ways. And this is of not only practical interest, but also scientific interest, since the application of LLMs significantly changes both individual aspects of activity and entire industries.

In recent years, there has been a lot of research that has tried to estimate LLMs [1-8] in different ways, but all of them face the same issues:

- Extremely rapid evolution makes any research quickly outdated.
- Choosing a good estimation dataset is extremely hard, since we don’t know, what data was used to train these models. And most of the proposed datasets have become biased, since the next generation of models will very likely be fine-tuned on them.

- Choosing the right estimation criteria and metrics that represent them is also challenging. That's why most researchers choose different metrics or calculate multiple metrics for the same experiments.
- Model constraints limit or make independent research very costly. As a result, researchers tend to use limited datasets [2-5] or prefer open-weighted local or older free models [1-2, 4-5], which often cannot compete with cloud-based counterparts.

In addition to these issues, most research focuses on specific domains, such as medicine [1, 5], education [3], market research [6-8], etc. So, their results and datasets cannot be directly generalized to broad-domain LLM estimation. And even papers that don't focus on specific domains [2, 4] use specific data representations, such as role-play dialogues [2] or multiple responses and user ratings [4].

Moreover, most research focuses on ground-truth estimation [1-5], comparing model responses with predetermined answers that are usually not generated and structurally don't correspond to LLM responses. Such a comparison requires more sophisticated preprocessing and/or postprocessing. Preprocessing may include prompting the model to limit its answers and make them more structurally similar to the expected answers. Postprocessing requires more complex embedding methods to improve precision and reduce sensitivity to structural differences between the generated and expected answers.

We also conduct ground-truth estimation in our work, but our main assumption is based on cross-model STS. We hypothesize that models with high inter-model semantic similarity may produce responses that are semantically close under the selected benchmark and evaluation metrics but may differ in request latency, context size, input price, and output price. So, identifying groups of semantically similar models may support selecting a more efficient model on these metrics while maintaining relatively similar responses.

Thus, the reviewed literature suggests that existing studies rarely combine reference-based semantic adequacy, inter-model semantic similarity, and operational indicators within a single comparative framework. This gap defines the scientific task of this study: to develop and test a comparative framework for assessing the STS between different LLMs and their operating modes, using a common FAQ-based benchmark, multiple STS metrics, and operational indicators such as latency and token cost.

MATERIALS AND METHODS

Understanding how close two texts are to each other is a classical problem in natural language processing (NLP). However, even now, choosing the best metric remains an open question, and in many cases, different researchers can propose their own metrics to estimate text similarity [4, 16-22]. But different metrics may use different approaches, so it is important to understand exactly how each metric is calculated.

Two main approaches to measuring text similarity are lexical and semantic. The lexical similarity measures the degree to which the word sets of two given languages or texts are similar. The semantic similarity measures the degree to which the meanings of two text fragments are similar, rather than just matching words [11-12]. While lexical similarity is easier to calculate and interpret, it focuses solely on a text's lexical structure, ignoring its meaning. Therefore, lexical similarity metrics usually perform poorly when the structures of the texts being compared differ.

In contrast, semantic similarity is less sensitive to text structure and focuses on overall text meaning, allowing comparisons of texts with different structures and lengths that cover the same topic. But at the same time, semantic similarity is often confused with semantic relatedness [12], which measures the relationship between two words or concepts based on the closeness of their meanings [13]. For this reason, semantic similarity is more specific and can be considered as a particular case of semantic relatedness. However, semantic similarity estimation is more complex and requires additional text processing to extract

meaning. This makes semantic similarity estimation more challenging and requires more sophisticated methods.

There are a lot of different metrics that can be used for semantic estimation [12, 14-15], but in general, they can be divided into three categories:

- Corpus-based [12] or structure-based [14] metrics;
- Knowledge-based [12] or feature-based [14] metrics;
- Hybrid [14] metrics.

Though the boundaries between groups aren't always clear, and some research [15] doesn't distinguish between hybrid and feature-based metrics, each metric has its own advantages. After analyzing prior studies [4, 13-22], we selected three metrics that are suitable for our experimental purpose:

- Latent Semantic Analysis (LSA) [23] using the TF-IDF vectorization [24].
- Cosine similarity based on Sentence-BERT (SBERT) [18] embeddings;
- BERTScore [19].

Latent Semantic Analysis (LSA) [23] is a traditional natural language processing (NLP) technique that uses singular value decomposition (SVD) [25-27] to identify latent patterns and relationships between a set of documents and the terms and concepts they contain. As input to SVD, we provide text vectors generated using the TF-IDF (term frequency–inverse document frequency) vectorizer. For our implementation, we used truncated SVD [26] with 10 components.

Cosine similarity [12] is one of the most widely used methods for measuring similarity. For this, we convert text into high-dimensional vectors that capture context using Sentence-BERT (SBERT) [18]. Then we calculate the cosine of the angle between vectors, which measures similarity regardless of text length:

$$similarity = cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}.$$

BERTScore [19] computes similarity using the contextualized token embeddings from the BERT model. It captures deep semantic nuances by matching tokens in the candidate and reference sentences rather than relying on surface-level overlaps.

In our estimation framework, we estimate two types of STS:

- Ground-truth STS score that measures the similarity between the ground-truth answer from the MFAQ subset and the answer generated by some LLM.
- Cross-model STS score that measures the similarity between answers generated by different LLMs.

While the ground-truth STS score provides a baseline based on the expected answer and allows evaluating how close the generated response is to the commonly provided answer, the cross-model STS score shows how close the answers of different LLMs are.

Combining these two perspectives allows us to compare models both by semantic proximity to reference responses and by semantic similarity between model outputs, together with descriptive latency and cost indicators.

RESULTS AND DISCUSSION

As mentioned earlier, we selected the 1000-pair English subset from the validation dataset of the Multilingual FAQ Dataset (MFAQ) [9] as the estimation dataset. The original English validation dataset contains 151825 question-answer pairs across 8178 unique web domains. To form our subset, we downloaded the corresponding Parquet file for the English validation dataset from Hugging Face (see the MFAQ official repository for more details) and randomly chose 1000 question-answer pairs. For random generation, we read the Parquet file as a pandas DataFrame in Python and used the DataFrame's built-in *sample*

method with *random_state* = 42 to select 1000 records. As a result, our estimation subset contained 1000 question-answer pairs from 883 unique web domains.

For ground-truth STS metrics, we calculate the LSA (Latent Semantic Analysis) score, the Cosine Similarity score, the BERTScore and the latency for each model, comparing its answers to the ground-truth answers from the MFAQ dataset. Cross-model STS metrics are estimated as the LSA score, the Cosine Similarity score, and the BERTScore for each pair of models. Additionally, to compare model costs and available context, we obtained the context size and the price in USD for model input and output per 1M tokens from the Price Per Token site [28] (see **Table 2**).

Table 2. **Context size and model input/output prices per 1M tokens for the researched models from the Price Per Token site [28]**

LLM	Context size	Input price, USD	Output price, USD
GPT-5-mini	400K	0.25	2.00
GPT-4.1	1.0M	2.00	8.00
Claude Haiku 4.5	200K	1.00	5.00
Claude Sonnet 4.6	1.0M	3.00	15.00
Claude Opus 4.6	1.0M	5.00	25.00

After running our estimation framework for each selected model, we grouped our results for ground-truth STS metrics by the average request latency (see **Table 3**), context size (see **Table 4**), input price (see **Table 5**), and output price (see **Table 6**).

Table 3. **The ground-truth STS metrics for the researched models, depending on the average request latency (in seconds)**

Model	Latency, s	Cosine Similarity Score	BERTScore	LSA Score
GPT-5-mini	25.677	0.5934	0.8267	0.6259
GPT-4.1	10.233	0.6060	0.8451	0.6439
Claude Haiku 4.5	7.359	0.4197	0.8145	0.4570
Claude Sonnet 4.6	7.863	0.4810	0.8280	0.5027
Claude Opus 4.6	8.310	0.4505	0.8252	0.4658

From the results obtained, we observe that some models from different providers achieve comparable reference-based STS scores. This may indicate partially similar semantic response patterns on the selected benchmark. However, no conclusions can be drawn about internal optimization priorities from these results alone. Interestingly, BERTScore places GPT-5-mini between Claude Sonnet 4.6 and Claude Opus 4.6 and reports lower STS than in GPT-4.1 (see **Table 3**).

Also, as shown in **Table 3**, the average request latency for GPT-5-mini is much higher than for other models, especially GPT-4.1, the larger model from the same provider. It may seem contradictory, since GPT-5-mini has much less context size and, intuitively, should

Table 4. **The ground-truth STS metrics for the researched models, depending on the context size in thousands**

Model	Context size, K	Cosine Similarity Score	BERTScore	LSA Score
GPT-5-mini	400	0.5934	0.8267	0.6259
GPT-4.1	1000	0.6060	0.8451	0.6439
Claude Haiku 4.5	200	0.4197	0.8145	0.4570
Claude Sonnet 4.6	1000	0.4810	0.8280	0.5027
Claude Opus 4.6	1000	0.4505	0.8252	0.4658

Table 5. **The ground-truth STS metrics for the researched models, depending on the input price per 1M tokens in USD**

Model	Input price, USD	Cosine Similarity Score	BERTScore	LSA Score
GPT-5-mini	0.25	0.5934	0.8267	0.6259
GPT-4.1	2.00	0.6060	0.8451	0.6439
Claude Haiku 4.5	1.00	0.4197	0.8145	0.4570
Claude Sonnet 4.6	3.00	0.4810	0.8280	0.5027
Claude Opus 4.6	5.00	0.4505	0.8252	0.4658

Table 6. **The ground-truth STS metrics for the researched models, depending on the output price per 1M tokens in USD**

Model	Output price, USD	Cosine Similarity Score	BERTScore	LSA Score
GPT-5-mini	2.00	0.5934	0.8267	0.6259
GPT-4.1	8.00	0.6060	0.8451	0.6439
Claude Haiku 4.5	5.00	0.4197	0.8145	0.4570
Claude Sonnet 4.6	15.00	0.4810	0.8280	0.5027
Claude Opus 4.6	25.00	0.4505	0.8252	0.4658

be faster. We manually investigated these two models and compared the results. First of all, we compared additional model properties using the Price Per Token site [28] and found that GPT-5-mini has much higher latency in generating the first output token, as measured

by Time to First Token (TTFT). For GPT-4.1, TTFT is only 0.61 s; for GPT-5-mini, it is 14.39 s. Additionally, the output speed for token generation in GPT-4.1 is higher than in GPT-5-mini, at 109 tokens/second, compared to 77 tokens/second in GPT-5-mini.

After manual investigation of the results, we also found that GPT-5-mini tends to generate longer responses, which, in combination with higher TTFT and lower token generation speed, results in approximately 2.51 times the latency of GPT-4.1.

Another interesting insight is that the more advanced and costly Claude Opus 4.6 provides semantically less similar answers compared to the cheaper Claude Sonnet 4.6. Although the reason for this requires further investigation, one possible explanation is that longer or more elaborate responses may receive lower similarity scores than shorter FAQ-style reference answers; however, this assumption requires additional verification.

Moreover, the comparison with the context size (see **Table 4**) clearly indicates that it does not reveal a clear relationship between the STS estimation and models with the same context size can have very different STS scores, and even some models with smaller context windows achieved higher STS scores than certain models with larger context windows in our experiment.

The input and output price comparison (see **Table 5-6**) shows that a high cost for model tokens doesn't guarantee high STS scores, and that using relatively less expensive models, such as GPT-4.1 and GPT-5-mini, can yield better STS than more expensive models.

The cross-model STS metrics are displayed as heat maps for each metric to visualise similarities across models (see **Fig. 1-3**).

The cross-model STS analysis yields several informative observations, especially regarding the metrics that can be used to estimate it. Our experiments show that, in the selected setup, BERTScore has limited discriminative power for cross-model comparison, since it produces very similar values for many model pairs (see **Fig. 2**). These results make it hard to use BERTScore to estimate cross-model STS, since these differences aren't practically meaningful or statistically significant and can be misinterpreted.

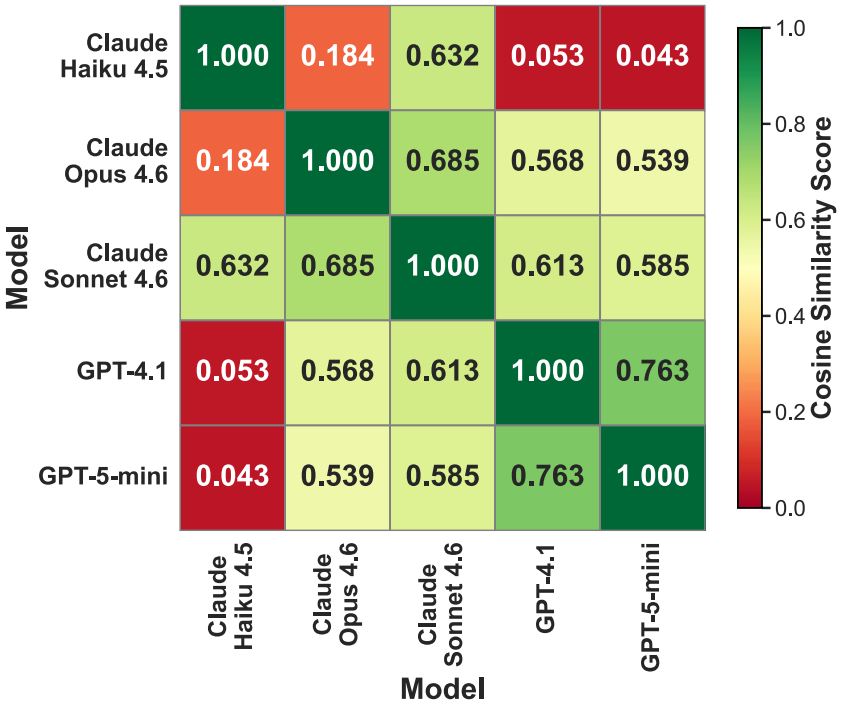

Fig. 1. The cross-model STS estimation using the Cosine Similarity Score

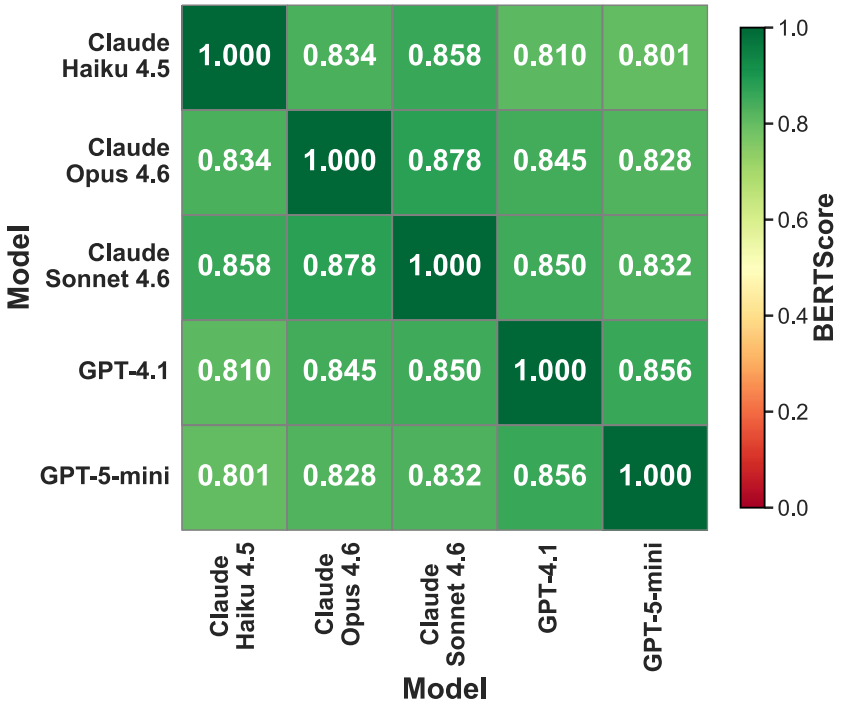

Fig. 2. The cross-model STS estimation using the BERTScore

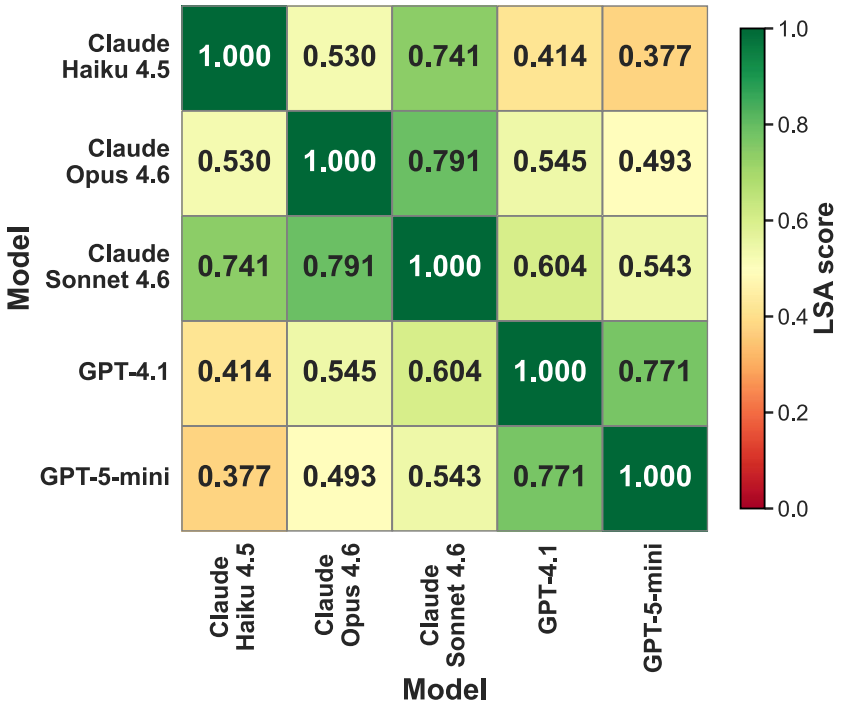

Fig. 3. The cross-model STS estimation using the LSA Score

In contrast, cosine similarity provides a wider spread of pairwise estimates and more clearly separates model pairs with relatively high and low similarity (see **Fig. 1**). At the

same time, the LSA score provides an intermediate level of separation between model pairs, clearly showing the high and low similarity of the models (see **Fig. 3**).

If we analyse the results indicate high semantic similarity between the two OpenAI models under the selected metrics (see **Fig. 1-3**). Anthropic models also show high similarity among themselves, but we can clearly see that Claude Sonnet 4.6 occupies an intermediate position between Claude Haiku 4.5 and Claude Opus 4.6 and is highly similar to both. However, the comparison between Claude Haiku 4.5 and Claude Opus 4.6 indicates lower semantic similarity than for other within-provider model pairs.

CONCLUSION

STS is an important metric for estimating the similarity of two texts, especially when we consider AI-generated texts and the variety of LLMs that can generate them. In this work, we presented the results of STS estimation for 5 LLMs from two providers (Anthropic and OpenAI). For our research, we developed the estimation framework based on the 1000-pair English subset from the MFAQ validation dataset.

In contrast to many reference-based studies, we estimated not only reference-based STS scores but also cross-model STS scores by comparing models. This extends the reviewed approaches by adding cross-model STS analysis, which provides additional insight into inter-model response similarity. However, as shown in the work, it strongly depends on the metric used, since different metrics yield different sensitivities for cross-model STS estimation. In our experimental setup, the LSA score and cosine similarity were the most informative metrics for cross-model comparison, whereas BERTScore yielded less sensitive and interpretable results.

Also, our results suggest that, in this experimental setting, models from the same provider are more semantically similar to each other than to models from another provider. We also showed that similar ground-truth estimates don’t necessarily imply high STS between the models, as evidenced by the cross-model STS scores.

Additionally, our experimental results showed that more expensive models did not consistently achieve the highest STS scores on the selected benchmark. Therefore, for the benchmark considered, some medium-cost or relatively inexpensive models may offer a reasonable balance between STS-based indicators, latency, and cost.

Despite the results obtained, the STS of AI-generated texts still requires further research using larger datasets and more models to provide a more complete picture. Also, in this research, we minimized any additional prompting from our side, but it can significantly affect the responses generated by models.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and/or publication of this paper.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [V.P., O.C., O.H., R.S.]; methodology, [V.P., O.C., O.H.]; software, [V.P., O.C.]; validation, [V.P., O.C., O.H.]; formal analysis, [V.P., O.C., O.H.].; investigation, [V.P., O.C., O.H.]; resources, [V.P., O.C., O.H.]; data curation, [V.P., O.C., O.H.]; writing –

original draft preparation, [V.P., O.C.]; writing – review and editing, [V.P., O.C., O.H., R.S.]; visualization, [V.P., O.C.] supervision, [V.P., R.S.]; project administration, [V.P., R.S.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Sblendorio, E., Lo Cascio, A., Napolitano, D., Germini, F., Dentamaro, V., Piredda, M., & Cicolini, G. (2025). Assessing and Comparing Free Large Language Models' Responses to a Clinical Case: Accuracy, Safety, and Reliability. *Computational Intelligence Methods for Bioinformatics and Biostatistics*, CIBB 2024. Lecture Notes in Computer Science, 15276, 134-149. https://doi.org/10.1007/978-3-031-89704-7_11

[2] Lu, D., Jeuring, J., & Gatt, A. (2025). Evaluating LLM-Generated Versus Human-Authored Responses in Role-Play Dialogues. In *Proceedings of the 18th International Natural Language Generation Conference*, Association for Computational Linguistics, Hanoi, Vietnam, 20–40. https://doi.org/10.48550/arXiv.2509.17694

[3] Ding, X. et al. (2024). Evaluation of LLMs and Other Machine Learning Methods in the Analysis of Qualitative Survey Responses for Accessible Engineering Education Research. *2024 ASEE Annual Conference & Exposition*, Portland, Oregon, USA, 1-24. https://doi.org/10.18260/1-2--47360

[4] Wu, X., Lin, W., Akgul, O., & Bauer, L. (2025). Estimating LLM Consistency: A User Baseline vs Surrogate Metrics. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics*, Suzhou, China, 30530-30544. https://doi.org/10.18653/v1/2025.emnlp-main.1554

[5] Lee, P. C., Sharma, S. K., Motaganahalli, S., & Huang, A. (2023). Evaluating the Clinical Decision-Making Ability of Large Language Models Using MKSAP-19 Cardiology Questions. *JACC: Advances*, 2(9). https://doi.org/10.1016/j.jacadv.2023.100658

[6] Zhang, L., Wang, J., Wu, J., & Zhang, Z. (2026). RetailBench: Evaluating Long-Horizon Autonomous Decision-Making and Strategy Stability of LLM Agents in Realistic Retail Environments. *ArXiv preprint*. https://doi.org/10.48550/arXiv.2603.16453

[7] Brand, J., Israeli, A., & Ngwe, D. (2023). Using LLMs for Market Research. *Harvard Business School Working Paper*, No. 23-062, April 2023 (Revised October 2025). https://doi.org/10.2139/ssrn.4395751

[8] Wang, M., Zhang, D., & Zhang, H. (2024). Large Language Models for Market Research: A Data-augmentation Approach. *Social Science Research Network* (SSRN). https://doi.org/10.2139/ssrn.5057769

[9] De Bruyn, M., Lotfi, E., Buhmann, J., & Daelemans, W. (2021). MFAQ: a Multilingual FAQ Dataset. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering, Association for Computational Linguistics*, Punta Cana, Dominican Republic, 1-13. https://doi.org/10.18653/v1/2021.mrqa-1.1.

[10] OpenAI. Introducing ChatGPT. *OpenAI*, November 30, 2022. Retrieved from: https://openai.com/index/chatgpt/

[11] Resnik, P. (1999). Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11, 95-130. https://doi.org/10.1613/jair.514

[12] Harispe, S., Ranwez, S., Janaqi, S., & Montmain, J. (2015). *Semantic Similarity from Natural Language and Ontology Analysis. Synthesis Lectures on Human Language*

Technologies Series, 8(1). Springer Cham. ISBN: 978-3-031-01028-6. https://doi.org/10.2200/S00639ED1V01Y201504HLT027

[13] Feng, Y., Bagheri, E., Ensan, F., & Jovanovic, J. (2017). The state of the art in semantic relatedness: a framework for comparison. *Knowledge Engineering Review*, 32(10), 1-30. https://doi.org/10.1017/S0269888917000029

[14] Slimani, T. (2013). Description and Evaluation of Semantic Similarity Measures Approaches. *International Journal of Computer Applications*, 80, 25-33. https://doi.org/10.5120/13897-1851

[15] Haque, S., Eberhart, Z., Bansal, A., & McMillan, C. (2022). Semantic similarity metrics for evaluating source code summarization. In *Proceedings of the 30th IEEE/ACM International Conference on Program Comprehension (ICPC '22)*, Association for Computing Machinery, New York, NY, USA, 36–47. https://doi.org/10.1145/3524610.3527909

[16] Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R. St., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Strope, B., & Kurzweil, R. (2018). Universal Sentence Encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics*, Brussels, Belgium, 169-174. https://doi.org/10.18653/v1/D18-2029

[17] Conneau, A., Kiela, D., Schwenk, H., Barrault, L., & Bordes, A. (2018). Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. *ArXiv preprint*. https://doi.org/10.48550/arXiv.1705.02364

[18] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3973-3983. https://doi.org/10.18653/v1/D19-1410

[19] Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating Text Generation with BERT. *International Conference on Learning Representations*, 1-43. https://doi.org/10.48550/arXiv.1904.09675

[20] Fu, J., Ng, S.-K., Jiang, Z., & Liu, P. (2023). GPTScore: Evaluate as You Desire. *ArXiv preprint*. https://doi.org/10.48550/arXiv.2302.04166

[21] Manakul, P., Liusie, A., & Gales, M. J. F. (2023). SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Singapore, 9004-9017. https://doi.org/10.18653/v1/2023.emnlp-main.557

[22] Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., & Zhu, C. (2023). G-Eval: NLG Evaluation using Gpt-4 with Better Human Alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Singapore, 2511-2522. https://doi.org/10.18653/v1/2023.emnlp-main.153

[23] Dumais, S. T. (2004). Latent Semantic Analysis. *Annual Review of Information Science and Technology*, 38, 188–230. https://doi.org/10.1002/aris.1440380105

[24] Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management: an International Journal*, 24(5), 513-523. https://doi.org/10.1016/0306-4573(88)90021-0

[25] Golub G. H., & Reinsch, C. (1970). Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5), 403-420. https://doi.org/10.1007/BF02163027
[26] Hansen, P. C. (1987). The truncated SVD as a method for regularization. *BIT Numerical Mathematics*, 27(4), 534-553. https://doi.org/10.1007/BF01937276
[27] Wall, M. E., Rechtsteiner, A., & Rocha, L. (2003). Singular Value Decomposition and Principal Component Analysis. In *A Practical Approach to Microarray Data Analysis*, 5, 91-109. https://doi.org/10.1007/0-306-47815-3_5
[28] Ventures, LLC (2026). LLM API Pricing 2026 – Compare 300+ AI Model Costs. Price Per Token. https://pricepertoken.com/

ОЦІНКА СЕМАНТИЧНОЇ ПОДІБНОСТІ ТЕКСТУ РІЗНИХ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

Віталій Парубочий*  , ***Олександр Чмихало, Олег Гусак, Роман Шувар***
Кафедра системного проектування,
Львівський національний університет імені Івана Франка,
вул. М. Драгоманова, 50, Львів, 79005, Україна

АНОТАЦІЯ

Вступ. Оцінка семантичної подібності текстів (СПТ) є одним із корисних підходів на основі проксі-методів для аналізу семантичних відмінностей між відповідями, згенерованими різними великими мовними моделями (ВММ), що використовуються в чатах штучного інтелекту та агентами штучного інтелекту. Різні дослідження останніх років намагалися оцінити СПТ для відповідей, згенерованих штучним інтелектом, але вони зазвичай зосереджені на оцінці істинності шляхом порівняння відповідей, згенерованих штучним інтелектом, із заздалегідь визначеними відповідями, які в більшості випадків структурно відрізняються від згенерованих текстів. Однак, міжмодельна оцінка СПТ може надати цінну інформацію про подібність моделі, незалежно від того, наскільки добре вони узгоджуються із заздалегідь визначеними відповідями.

Матеріали та методи. У цій статті ми оцінюємо обидва типи СПТ, але, в першу чергу, зосереджуємося на СПТ між відповідями, згенерованими кількома вибраними ВММ від різних постачальників. Ми будуємо нашу систему оцінювання, використовуючи підмножину часто задаваних питань (ЧЗП, англ. FAQ – Frequently Asked Questions) англійською мовою з набору даних MFAQ, оскільки вона пропонує широкий спектр тем та чітку, просту у використанні структуру.

Результати. Результати наших експериментів складаються з двох частин. У першій частині ми оцінюємо базові показники СПТ, використовуючи кілька метрик (косинусна подібність, BERTScore та латентно-семантичний аналіз (ЛСА)), щоб проаналізувати, наскільки близько відповіді, згенеровані кожною моделлю, співвідносяться з базовими відповідями набору даних MFAQ. У другій частині ми оцінюємо міжмодельні показники СПТ, використовуючи ті самі метрики, щоб проаналізувати, наскільки близько моделі узгоджуються одна з одною. Крім того, ми вимірюємо середню затримку запиту та використовуємо загальнодоступну інформацію про розмір контексту моделі та ціни вхідних і вихідних даних, щоб дослідити можливі зв'язки між цими параметрами та показниками якості на основі СПТ.

Висновки. Наші результати вказують на відносно високу семантичну подібність між моделями від одного постачальника у вибраній експериментальній установці та

демонструють, що високі показники СПТ, що відповідають реальним даним, не обов'язково означають високу міжмодельну семантичну подібність у межах вибраного тесту оцінки та метрик, як видно з аналізу міжмодельних показників СПТ. Крім того, ми показуємо, що не всі метрики однаково придатні для міжмодельної оцінки СПТ, і що такі метрики, як косинусної подібності або ЛСА, можуть забезпечити кращу чутливість, ніж BERTScore або подібні метрики.

Ключові слова: семантична подібність тексту (СПТ), велика мовна модель (ВММ), набір даних MFAQ, косинусна подібність, BERTScore, латентно-семантичний аналіз (ЛСА)

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
06 April, 2026	16 May, 2026	18 May, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 65–82
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.41

CLOUD-AGNOSTIC INTELLIGENT MONITORING ARCHITECTURE FOR CLIMATE ANOMALIES IN EARLY WARNING SYSTEMS

Vasyl Lyashkevych *, ***Yurii Marchuk***, ***Petro Kulyk***, ***Andrii Patynko***, ***Taras Kerod***
Department of System Design
Ivan Franko National University of Lviv,
50 Drahomanova Str., Lviv, 79005, Ukraine

Lyashkevych, V., Marchuk, Y., Kulyk, P., Patynko, A., & Kerod, T. (2026). Cloud-agnostic Intelligent Monitoring Architecture for Climate Anomalies in Early Warning Systems. *Electronics and Information Technologies*, 34, 65–82. https://doi.org/10.30970/eli.34.4

ABSTRACT

Background. Climate-related hazards require early warning systems to combine heterogeneous environmental observations, detect anomalous spatiotemporal patterns, and disseminate warning signals to stakeholders. Existing solutions separate software architecture, anomaly-detection models, and decision logic to transform deviations into early signals.

Materials and Methods. The study develops a cloud-agnostic intelligent monitoring architecture for early warning systems, supported by ML-based anomaly identification and decision support. A hierarchical anomaly model is introduced for climate zones, areas, and regions. It evaluates four types of evidence: deviation from normal state, consistency with neighboring regions, consistency with parent area, and the influence of contextual factors. The method combines region-specific detector training, multi-component anomaly scoring, persistence-based warning confirmation, and CEP-based escalation. A synthetic spatiotemporal climate graph was used to compare the Robust Z-score statistical method, the Isolation Forest anomaly-detection algorithm, the One-Class Support Vector Machine method, and the proposed HCOIM-EWS method, where HCOIM-EWS denotes Hierarchical Context-Orchestrated Intelligent Monitoring for Early Warning Systems.

Results and Discussion. The proposed architecture integrates ML-based detector selection, region-specific training, hierarchical anomaly scoring, CEP-based warning logic, and notification distribution. In simulation, HCOIM-EWS achieved the lowest false-alarm count while maintaining comparable event-level recall. Compared with Robust Z-score, Isolation Forest, and One-Class SVM, the proposed method reduced false alarms by 55.0%, 22.6%, and 49.7%. The method is suitable for warning-oriented intelligent monitoring, where operational reliability and false-alarm reduction are as important as point-level sensitivity.

Conclusion. The proposed method extends anomaly detection in early warning systems from isolated local time-series analysis to hierarchical, context-aware, and event-driven intelligent monitoring. This confirms the technical feasibility of the architecture and justifies its use as a reusable template for climate-risk monitoring platforms.

Keywords: anomaly detection, early warning systems, intelligent monitoring, cloud-agnostic architecture, complex event processing, machine learning.

INTRODUCTION

Early warning systems (EWSs) are key instruments of disaster risk reduction because they support timely monitoring, forecasting, communication, and coordinated response to hazardous events [1].

Anomaly detection identifies patterns that deviate from expected behavior, including outliers, novelties, exceptions, or potentially dangerous states [2, 3]. In environmental and meteorological settings, anomaly detection becomes a basis for identifying deviations that may precede floods, storms, droughts, heatwaves, or other natural hazards [3]. Recent ML-based studies also demonstrate the growing role of artificial intelligence in modeling extreme weather events, quality control of meteorological observations, and heavy-rainfall observation analysis [4-6].

This paper uses weather singularities as a motivating class of climate-related deviations and studies their identification through hierarchical anomaly monitoring. Regardless of terminology, the core problem is how to build a scalable system that ingests heterogeneous climate signals, synchronizes them in time and space, trains and deploys multiple anomaly detectors, and transforms anomaly evidence into actionable warning signals.

Recent studies show that EWSs are moving from isolated forecasting tools to modular, digital, and intelligent architectures [7], with an emphasis on interoperability, resilience, governance, and integration of analytical and distributive workflows [8, 9]. Their practical implementation increasingly relies on scalable big data processing, automated machine learning, and climate zone segmentation to support distributed model training and region-specific analysis [10–12]. At the same time, anomaly detection in environmental monitoring is evolving from purely local time series analysis to spatiotemporal and context-dependent formulations, where a local anomaly is interpreted in relation to neighboring regions and broader climate regimes [13–15].

In this study, intelligent monitoring is understood as a continuous process that combines data acquisition, anomaly identification, contextual interpretation, and warning-oriented decision support. Such a perspective is consistent with recent work on intelligent monitoring, optimization of analytical pipelines by machine learning and genetic algorithms, and scenario-based environmental modeling [16–18].

This work addresses the lack of an integrated cloud-agnostic intelligent monitoring architecture that combines ML-based anomaly identification, warning-oriented decision support, hierarchical anomaly modeling, and EWS reference architecture. The objective is to develop and evaluate a cloud-agnostic intelligent monitoring architecture supported by the author-defined HCOIM-EWS method, which connects ML-based anomaly detection, warning-oriented decision logic, and reusable deployment principles for early warning systems. In this study, HCOIM-EWS denotes Hierarchical Context-Orchestrated Intelligent Monitoring for Early Warning Systems and is not used as a standardized or officially established IT-industry term. The study demonstrates the feasibility of the method through simulation on a synthetic hierarchical climate graph. The main contributions are: (i) a formal hierarchical anomaly model for climate-anomaly monitoring; (ii) the author-defined HCOIM-EWS method for transforming anomaly candidates into warning-level evidence; (iii) a simulation-based comparison with three baseline anomaly-identification methods; and (iv) a cloud-agnostic reference architecture for early warning systems.

MATERIALS AND METHODS

The proposed solution assumes a hierarchically organized monitored space composed of climate zones, areas, and regions. For each region, the system maintains synchronized multivariate observations, local and neighboring context, and higher-level climatic aggregates. The event-driven cloud-agnostic architecture integrates distributed

APIs, Kafka, Big Data storage and preprocessing, AutoML training, and CEP-based warning fusion.

Problem formulation and decision-support objective

The problem addressed in this study is the detection of anomalous deviations associated with weather singularities that may serve as early indicators of natural hazards. Unlike conventional forecasting systems focused on isolated regional observations, the proposed method assumes that weather conditions in a given region may be influenced by patterns observed in neighboring regions and broader climatic domains. Accordingly, anomaly detection should not be restricted to a single local time series but should incorporate both vertical scaling across broader and narrower climatic levels and horizontal orchestration across adjacent regions.

Formal hierarchical anomaly model

Let R denote the set of monitored regions, A – set of climate areas, and Z – set of climate zones. For each region $r \in R$ and time moment t, the monitoring system observes a synchronized multivariate environmental vector $x_r(t) \in R^m$. Let $N(r)$ denote the set of horizontally neighboring regions and $P(r)$ denote the parent climatic area or zone associated with region r.

The normal behavior of a region is represented by the expected state vector $\widehat{x_r}(t)$, estimated from historical synchronized observations, temporal context, and higher-level climatic context. We define a regional anomaly as a deviation that simultaneously affects at least one of four dimensions: local deviation, horizontal coherence, vertical coherence, or contextual stress, which represents deviation of external environmental drivers from their expected baseline:

$$D_{loc}(r,t) = \left|\left|x_r(t) - \hat{x}_r(t)\right|\right|_{\Sigma_r^{-1}}, \tag{1}$$

where Σ_r is the covariance matrix of normal regional behavior,

$$D_{hor}(r,t) = \frac{1}{|N(r)|}\sum_{n=1}^{N(r)} w_{rn}\left|\left|x_r(t) - x_n(t)\right|\right|_2, \tag{2}$$

where w_{rn} is the strength of spatial linkage between neighboring regions,

$$D_{ver}(r,t) = \left|\left|x_r(t) - g_{P(r)}(t)\right|\right|_2, \tag{3}$$

where $g_{P(r)}(t)$ is the aggregated climatic state of the parent area or zone,

$$D_{ctx}(r,t) = \left|\left|c_r(t) - \hat{c_r}(t)\right|\right|_2, \tag{4}$$

where $c_r(t)$ is a vector of contextual drivers and $\hat{c}_r$ is its expected baseline,

$$A(r,t) = \alpha \cdot D_{loc}(r,t) + \beta \cdot D_{hor}(r,t) + \gamma \cdot D_{ver}(r,t) + \delta \cdot D_{ctx}(r,t), \tag{5}$$

where $\alpha + \beta + \gamma + \delta = 1$, and $\alpha, \beta, \gamma, \delta \geq 0$. A point is marked as a local anomaly candidate if $A(r,t) > \tau_r$; a warning is generated only after persistence and cross-region confirmation rules are satisfied within a sliding time window H.

The score is an operational monitoring indicator for warning-oriented decision logic, not a replacement for physical meteorological forecasting models.

Proposed HCOIM-EWS anomaly identification method

To operationalize the formal anomaly model, the author-defined method Hierarchical Context-Orchestrated Intelligent Monitoring for Early Warning Systems (HCOIM-EWS) is proposed. In this paper, HCOIM-EWS is used as the author's designation for a hierarchical context-orchestrated intelligent monitoring method for early warning systems, not as a standardized, generally accepted, or official IT-industry term. It integrates machine learning-based anomaly identification, hierarchical contextual interpretation, and warning decision rules. It considers:

- multi-source ingestion and synchronization of environmental data streams through APIs, Kafka topics, and storage services;
- construction of region-level, area-level, and zone-level feature spaces with explicit horizontal and vertical linking;
- construction and periodic retraining of region-specific anomaly detectors using synchronized multivariate feature vectors, baseline estimation, unsupervised anomaly scoring, and warning-oriented threshold calibration;
- online inference for each region with computation D_{loc}, D_{hor}, D_{ver}, and D_{ctx};
- aggregation of local anomaly events in the CEP layer to identify warning templates based on persistence, neighborhood support, and hierarchical escalation;
- dissemination of warning signals to the alert subsystem, dashboards, and external stakeholders.

Thus, HCOIM-EWS transforms ML-based anomaly candidates into operational warning events through hierarchical context and decision logic.

Machine learning model construction

The machine learning component was constructed as a region-specific anomaly identification pipeline. The objective was to adapt anomaly scoring to each region and interpret local results in spatial and hierarchical contexts.

For each monitored region r, the input data were represented as a synchronized multivariate environmental vector:

$$x_r(t) = \left[x_{r,1}(t),\ x_{r,2}(t),\ \ldots,\ x_{r,m}(t)\right], \tag{6}$$

where $x_r(t)$ describes synchronized environmental and contextual variables.

In the simulation, each region had 720 synchronized observations. These observations were used to estimate the baseline behavior of each region and to train or calibrate anomaly detectors.

The procedure included the stages summarized in **Table 1**. The ML component consists of a region-specific anomaly identification layer and a warning-oriented decision layer. The first layer detects candidate deviations, while the second layer determines whether these deviations are persistent, spatially supported, and sufficiently meaningful to generate an early warning. To clarify how the machine learning component was constructed, **Table 1** summarizes the model-building procedure used in the proposed method.

Table 1 shows that HCOIM-EWS extends point-level anomaly detection with horizontal, vertical, contextual, and CEP-based confirmation mechanisms.

Table 1. **Machine learning model-building procedure**

Stage	Input	Operation	Output
Data synchronization	Heterogeneous environmental observations	Time alignment, grouping by region, normalization	Synchronized regional feature vectors
Baseline estimation	Historical regional observations	Estimation of expected regional behavior	Region-specific normal behavior model
Baseline detector construction	Normalized feature vectors	Robust Z-score method, Isolation Forest algorithm, and One-Class SVM method	Comparative anomaly scores
Author-defined HCOIM-EWS score construction	Regional, neighboring, parent-level, and contextual features	Computation of D_{loc}, D_{hor}, D_{ver}, D_{ctx}	Aggregated anomaly score $A(r,t)$
Thresholding	Anomaly scores	Quantile-based threshold selection	Local anomaly candidates
Warning decision	Local anomaly candidates	CEP rules, persistence check, spatial and hierarchical confirmation	Warning-level anomaly events

Cloud-agnostic EWS workflow

At the application level, the proposed early warning system is organized as a pipeline that transforms heterogeneous environmental data into warning-oriented information products. The workflow begins with data acquisition and normalization, continues with regional forecasting and anomaly detection, and ends with complex event processing and signal dissemination to end users and response services.

Simulation design

Anomalies that appear weak or noisy at one spatial scale may become interpretable when neighboring regions and parent climatic structures are considered together. Vertical scaling moves from smaller territorial units toward larger climatic aggregations and back, allowing the monitoring system to assess whether local instability is consistent with broader climatic states.

Climate regimes motivate the hierarchical grouping of regions. The Köppen–Geiger classification supports this conceptual partitioning and justifies region-specific models.

Horizontal and vertical orchestration evaluate the target region along with neighboring regions for anomaly interpretation. This is important when the local signal needs to be amplified, attenuated, or contextualized using spatial coherence (**Fig. 1**).

Fig. 1 shows how the target region is compared with adjacent regions. If the target region deviates while neighboring regions remain stable, the system may treat the signal as a local anomaly candidate. If the deviation is also supported by neighboring regions, the anomaly evidence becomes stronger and may contribute to warning-level escalation.

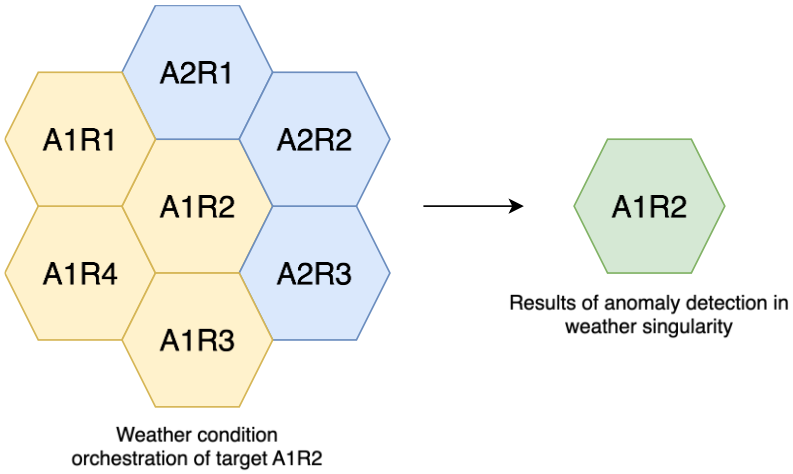

Fig. 1. Horizontal orchestration of a target region and neighboring regions for spatial coherence assessment. Source: developed by the authors

To test the feasibility of the anomaly model and the proposed monitoring method, a synthetic spatiotemporal climate graph was generated for 36 regions organized into 3 zones × 3 areas × 4 regions. Each regional time series contained 720 synchronized observations and combined seasonal structure, local trend, global climatic oscillation, zone-specific variation, area-specific variation, and random noise.

In this study, the synthetic spatiotemporal climate graph is understood as an artificially generated hierarchical model of climate zones, areas, and regions. Each region is represented as a graph node with a synchronized environmental time series, while links between nodes describe spatial neighborhood relations and membership in higher climatic levels. This representation allows controlled generation of normal states, injected anomalies, and ground-truth anomaly episodes for comparative evaluation of anomaly-identification methods.

Four anomaly types were injected into the synthetic environment:

- local spikes;
- regional drifts affecting all regions inside an area;
- hierarchy breaks where one area diverged from its zone-level regime;
- latent-driver anomalies affecting multiple semantically related regions.

In total, 33 anomaly episodes were injected and used as ground truth.

Four approaches were compared: the Robust Z-score statistical method, the Isolation Forest anomaly-detection algorithm, the One-Class SVM method, and the proposed author-defined HCOIM-EWS intelligent monitoring method.

All diagrams, workflows, algorithms, and architecture figures presented in this paper were redrawn and unified by the authors. Figures based on external concepts are explicitly marked as “developed by the authors based on [source]”, while fully original figures are marked as “developed by the authors”.

RESULTS AND DISCUSSION

Anomaly model simulation

The simulation inspected whether the model distinguishes isolated local deviations from spatially and hierarchically meaningful warning patterns. The synthetic climate graph contained 36 monitored regions arranged into 3 zones, 9 areas, and 36 regions. For each region, a synchronized multivariate time series of 720 observations was generated. Each observation included seasonal variation, local noise, regional trend, zone-specific climatic

oscillation, area-level variation, and contextual drivers. **Table 2** summarizes the main configuration parameters of the synthetic simulation environment.

Table 2. **Configuration of the synthetic simulation environment**

Parameter	Value used in simulation	Purpose
Number of climatic zones	3	Representation of broad climatic regimes
Number of climatic areas	9	Intermediate aggregation level
Number of regions	36	Region-specific anomaly detection
Regions per area	4	Local spatial neighborhood construction
Observations per region	720	Synchronized time-series simulation
Total regional observations	25,920	Complete synthetic monitoring dataset
Number of input dimensions	6	Multivariate environmental and contextual representation
Injected anomaly episodes	33	Ground truth for event-level validation
Compared methods	4	Robust Z-score method, Isolation Forest algorithm, One-Class SVM method, HCOIM-EWS method
Evaluation unit	Anomaly episode	Warning-level rather than point-level validation

The generated observations were used to compute four partial anomaly components: local deviation, horizontal deviation, vertical deviation, and contextual stress. The local deviation component reacted to abrupt changes in a single regional signal. The horizontal component evaluated whether the target region remained coherent with neighboring regions. The vertical component measured whether the region was consistent with its parent climatic area or zone. The contextual component represented abnormal pressure caused by external or latent drivers. The final anomaly score combined these components using weighted aggregation and a region-specific threshold estimated from baseline observations.

The injected anomalies were designed to activate different components of the proposed anomaly model. **Table 3** shows the relationship between anomaly type, spatial scope, expected dominant anomaly component, and warning-level interpretation.

The design supports component-level inspection because different anomaly classes should activate different evidence patterns.

The simulation confirmed that the four-component anomaly model produces more interpretable warning evidence than a purely local detector. Local spikes were mainly reflected by the local deviation component. Regional drifts increased both local and horizontal components. Hierarchy breaks were most clearly reflected by vertical incoherence. Latent-driver anomalies increased contextual stress and produced spatially distributed but semantically coherent deviations. Therefore, the model can explain not only that an anomaly occurred, but also which type of spatial, hierarchical, or contextual inconsistency contributed to the warning signal.

Table 3. **Injected anomaly types and expected response of the proposed anomaly model**

Anomaly type	Spatial scope	Expected dominant component	Warning-level interpretation
Local spike	Single region	Local deviation, D_{loc}	Short, abrupt regional disturbance
Regional drift	All regions inside one area	Local deviation and horizontal coherence, D_{loc} + D_{hor}	Gradual area-level instability
Hierarchy break	One area diverges from its zone	Vertical coherence, D_{ver}	Inconsistency between the area and the parent climatic regime
Latent-driver anomaly	Several semantically related regions	Contextual stress and horizontal coherence, D_{ctx} + D_{hor}	Distributed anomaly caused by hidden contextual pressure

The most important observation is that not every detected point anomaly should be escalated into an early warning. In the proposed model, an event is escalated only when the anomaly score remains persistent within a sliding window and is supported by neighboring regions, parent climatic structures, or contextual drivers. This makes the model suitable for early warning systems, where excessive sensitivity to isolated outliers may produce many false alarms and reduce trust in the warning service.

The model was inspected at the component level to check whether each component contributes distinct anomaly evidence.

The local deviation component was sensitive to abrupt single-region changes and therefore provided fast initial detection. However, this component alone was not sufficient for warning-level decisions because local noise, sensor artifacts, or short-term weather fluctuations may also produce high local anomaly scores. The horizontal coherence component reduced this limitation by comparing the target region with neighboring regions. If the deviation was not supported by spatial context, the probability of escalation decreased. This behavior is important for suppressing spatially inconsistent noise. An ablation-style inspection showed the expected behavior of the model (**Table 4**).

Table 4. **Component-level inspection of the proposed anomaly model**

Model variant	Included components	Expected strength	Expected limitation
Local-only model	D_{loc}	Fast reaction to abrupt regional deviations	High sensitivity to noise and isolated outliers
Local + horizontal model	D_{loc}+ D_{hor}	Suppression of spatially inconsistent false alarms	Weaker interpretation of hierarchy-level deviations
Local + horizontal + vertical model	D_{loc}+ D_{hor} + D_{ver}	Detection of inconsistencies between regions, areas, and zones	Limited explanation of hidden contextual drivers
Full HCOIM-EWS model	D_{loc} + D_{hor}+ D_{ver}+ D_{ctx}+ CEP rules	Balanced warning-level detection with reduced false alarms	Requires calibration of weights, thresholds, and CEP rules

The vertical coherence component evaluated whether regional behavior was consistent with the parent's climatic area or zone. This component was especially useful for hierarchy-break anomalies, where a local or area-level pattern diverged from the broader climatic regime. The contextual stress component captured abnormal influence from external or latent drivers. It was useful in cases where several regions changed in a coordinated way, but the change was not fully explained by local or neighboring time-series behavior.

The inspection confirms that each additional component contributes to a distinct type of monitoring evidence. The local component provides initial sensitivity, the horizontal component checks spatial support, the vertical component evaluates hierarchical consistency, and the contextual component captures external or latent pressure. CEP rules then transform the aggregated anomaly score into warning-level evidence.

Thus, the weighted anomaly score is a warning-oriented evidence function rather than a universal meteorological risk index.

Anomaly detection method implementation

The HCOIM-EWS method was implemented as a sequence of data-processing, model-training, inference, and warning-aggregation operations aligned with the proposed cloud-agnostic architecture (**Fig. 2**). The implementation assumes that each monitored region has its own synchronized data stream, historical baseline, neighboring regions, and parent climatic area or zone.

At the ingestion stage, heterogeneous environmental inputs are collected through distributed APIs and transformed into unified time-indexed observations. Missing or delayed values are handled through time alignment, caching of the last valid values, and normalization by input type. The resulting observations are written to the storage layer and also published as events for online inference.

At the feature-construction stage, the system builds three connected feature spaces: region-level features, neighborhood-level features, and parent-level climatic aggregates. Region-level features are used to estimate local deviation. Neighborhood-level features are used to estimate horizontal coherence. Parent-level aggregates are used to estimate vertical coherence. Contextual drivers are represented as an additional feature vector and compared with their expected baseline.

Fig. 2 shows that HCOIM-EWS separates local anomaly candidate generation from the final warning generation. This separation prevents local anomaly candidates from being treated as warnings without persistence and spatial support.

At the training stage, region-specific models are trained or calibrated on synchronized baseline observations. The Robust Z-score statistical method estimates robust statistical deviation, the Isolation Forest anomaly-detection algorithm learns isolation-based abnormality, the One-Class Support Vector Machine method estimates the boundary of normal behavior, and the proposed HCOIM-EWS intelligent monitoring method constructs a multi-component anomaly score from local, horizontal, vertical, and contextual evidence. In a production environment, this stage can be implemented through AutoML services that periodically retrain region-specific models and update thresholds.

At the online inference stage, each new observation is processed as follows:

- receive synchronized observation for region r at time t;
- compute local deviation D_{loc} using the regional baseline;
- compute horizontal deviation D_{hor} using neighboring regions $N(r)$;
- compute vertical deviation D_{ver} using the parent climatic aggregate $P(r)$;
- compute contextual stress D_{ctx} using contextual drivers;
- calculate the aggregated anomaly score $A(r,t)$;
- compare $A(r,t)$ with the region-specific threshold τ_r;
- publish a local anomaly event if the threshold is exceeded;

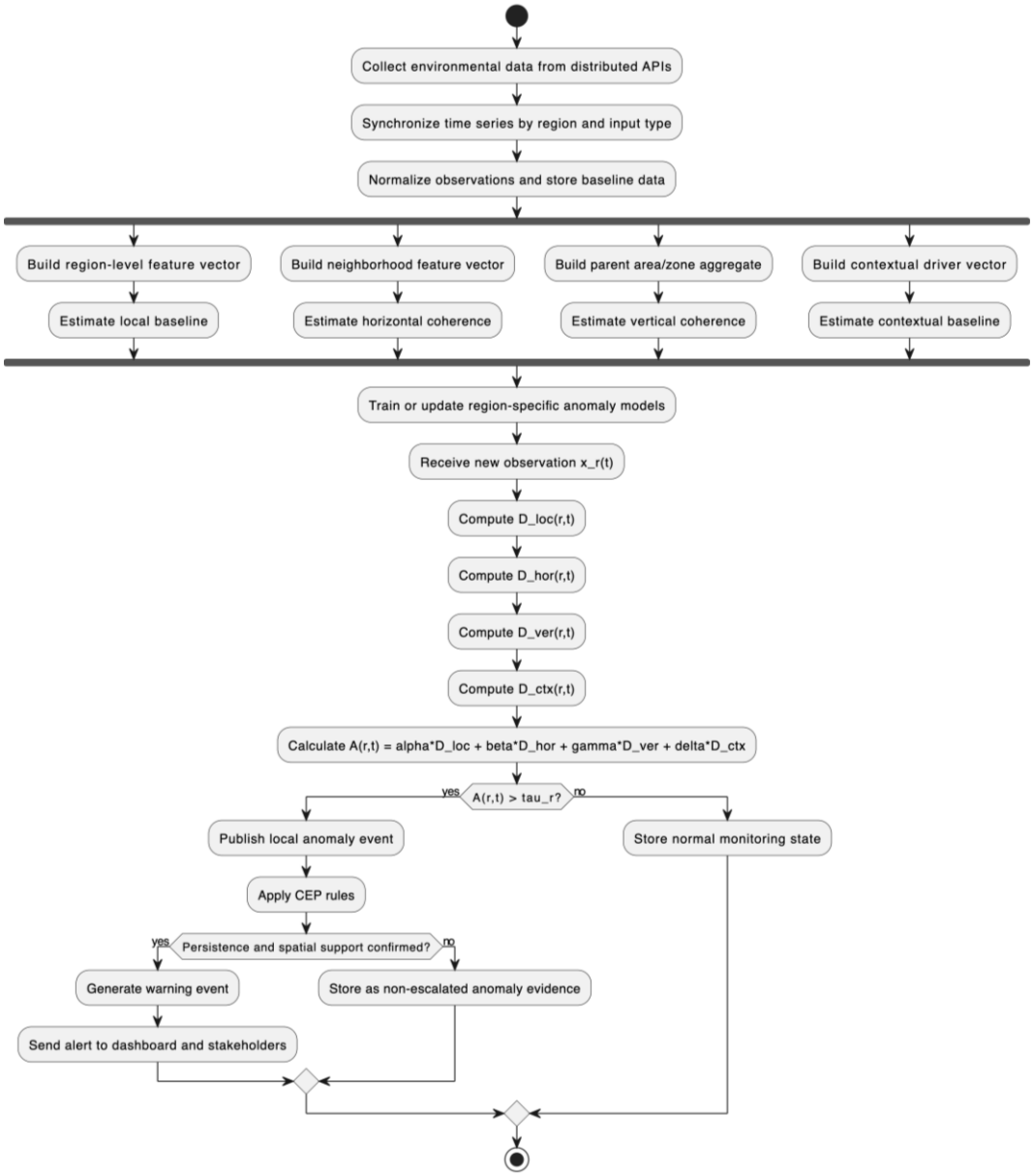

Fig. 2. Implementation flow of the HCOIM-EWS method from data ingestion to warning generation. Source: developed by the authors

- apply CEP rules for persistence, neighborhood support, and hierarchical escalation;
- generate a warning event only when the escalation conditions are satisfied.

The implementation separates ML-based anomaly candidate generation from warning-level decision-making. This separation is essential for early warning systems because a point anomaly is only a signal candidate, whereas a warning must be persistent, spatially supported, and operationally meaningful. As a result, HCOIM-EWS acts not only as an anomaly detector but also as an intelligent monitoring method that transforms local deviations into structured warning evidence.

Validation protocol

The validation protocol was designed to evaluate the proposed method from the viewpoint of warning management rather than only point-level anomaly classification. This distinction is important because an early warning system should not simply maximize the number of detected abnormal points. It should detect meaningful anomaly episodes while minimizing false alarms and maintaining acceptable detection delays. **Fig. 3** summarizes the validation procedure: synthetic graph generation, normal data simulation, anomaly injection, method comparison, and event-level evaluation.

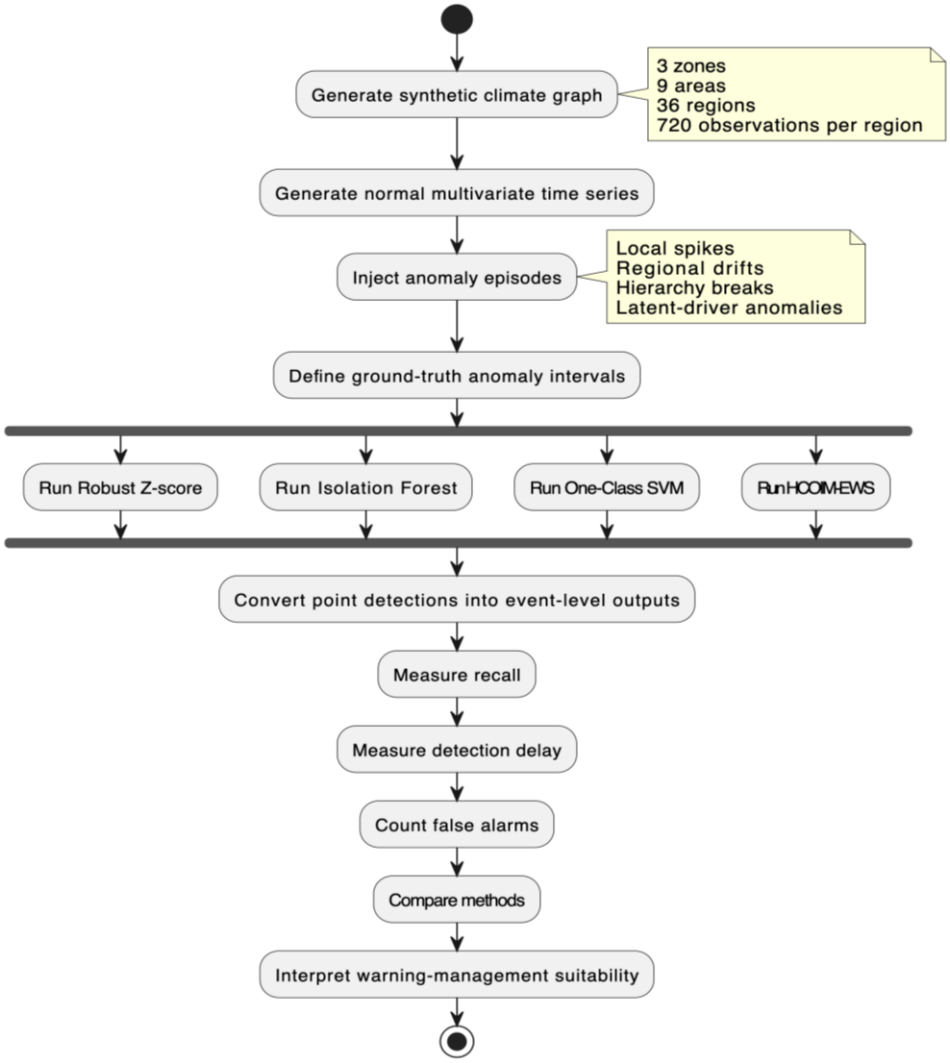

Fig. 3. Validation protocol for event-level evaluation of the HCOIM-EWS method. Source: developed by the authors

The main evaluation unit was an anomaly episode rather than an isolated anomalous point. An injected episode was counted as detected if at least one warning was generated within the allowed time window. Detection delay was calculated as the difference between the start of the injected episode and the first corresponding detection. False alarms were counted when the method generated warning-level events outside the ground-truth anomaly intervals.

HCOIM-EWS achieved the lowest false-alarm count while preserving a recall level comparable to the best baseline methods. This confirms that the proposed method is suitable for warning-oriented monitoring, where the operational cost of excessive false positives is high.

Simulation-based comparison of four methods

The compared methods were evaluated from the perspective of warning-level anomaly detection. Therefore, the evaluation focused not only on whether anomalous points were detected, but also on whether complete anomaly episodes were identified with acceptable delay and a limited number of false alarms. **Table 5** summarizes the threshold quantile used for each method, event-level recall, average detection delay, and the number of false alarms generated during the simulation.

Table 5. **Simulation results for the compared methods**

Compared approach	Threshold quantile	Event-level recall	Detection delay	False alarms
Robust Z-score	0.960	0.641	2.88	160
Isolation Forest	0.975	0.641	6.20	93
One-Class SVM	0.985	0.667	2.65	143
HCOIM-EWS	0.975	0.641	3.16	72

HCOIM-EWS achieved the lowest false-alarm count while preserving recall comparable to Robust Z-score and Isolation Forest. One-Class SVM had slightly higher recall but much more false alarms. Thus, the proposed method provides a more balanced warning-management profile, prioritizing stable and context-supported warning generation rather than sensitivity to isolated anomalous points. This behavior is especially important for early warning systems, where excessive false positives may reduce user trust and overload downstream response services.

The compared ML models were constructed on the same synchronized feature space, but they differed in the way anomaly evidence was interpreted. The Robust Z-score statistical method evaluated deviations from a robust statistical baseline. The Isolation Forest anomaly-detection algorithm identified observations that were easier to isolate in the feature space. The One-Class SVM method estimated the boundary of normal behavior. The proposed author-defined HCOIM-EWS intelligent monitoring method extended this point-level logic by adding spatial, hierarchical, and contextual interpretation.

To make the comparison more explicit, **Table 6** presents the relative false-alarm reduction achieved by HCOIM-EWS in comparison with the baseline approaches.

Table 6. **Relative improvement of HCOIM-EWS by false-alarm reduction**

Baseline approach	False alarms of baseline	False alarms of HCOIM-EWS	Absolute reduction	Relative reduction
Robust Z-score	160	72	88	55.0%
Isolation Forest	93	72	21	22.6%
One-Class SVM	143	72	71	49.7%

Thus, HCOIM-EWS mainly improves operational reliability by reducing false warnings. **Table 7** summarizes the operational interpretation of these results.

The method is therefore best interpreted as a balanced warning-management method, not as the most sensitive point detector.

Cloud-agnostic intelligent monitoring reference architecture for EWS

The proposed cloud-agnostic architecture treats anomaly identification and warning generation in EWSs as a sequence of distributed events and analytics operations rather than as a monolithic forecasting task. The monitored climate space is partitioned into zones, areas, and regions. Vertical and horizontal orchestration of those partitions allows the monitoring system to interpret local instability in a broader climatic context and to scale to hundreds of region-specific models.

Table 7. **Operational interpretation of model-evaluation results**

Evaluation aspect	Best-performing method	Interpretation
Lowest false-alarm count	HCOIM-EWS	Best warning-management reliability
Highest event-level recall	One-Class SVM	Highest sensitivity, but with high false-alarm burden
Shortest detection delay	One-Class SVM	Fastest detection in the simulation
Lowest delay among methods with low false alarms	HCOIM-EWS	Balanced compromise between delay and false-alarm reduction
Best operational trade-off	HCOIM-EWS	Most suitable for warning-oriented anomaly identification

The reference architecture consists of five logical layers: data acquisition, event and storage infrastructure, ML training and inference, warning decision support, and alert dissemination. In this interpretation, the architecture is not only a reference software design but also a practical implementation environment for intelligent monitoring: Kafka provides asynchronous event exchange, Spark-based processing supports scalable feature preparation, AutoML manages model training and retraining, and CEP converts anomaly evidence into warning-level decisions [10, 11].

Region-specific anomaly detectors publish local anomaly events that are aggregated by the context orchestration and CEP layers. Confirmed warning events are disseminated to dashboards, notification services, emergency services, and user-facing applications, reflecting the role of communication and response capacity in effective EWSs [19, 20].

Because environmental inputs differ in update periods, quality, and temporal resolution, the architecture performs time alignment, value reconciliation, caching, grouping, and normalization before training and inference.

The training contour includes historical data storage, feature preparation, model training, model registry, threshold calibration, and deployment to online inference services. Cloud-agnostic deployment is achieved by using provider-independent components such as containerized services, Kubernetes orchestration, Spark-compatible processing, object or distributed storage, and event streaming.

Fig. 4 presents the proposed reference architecture as an implementation template for HCOIM-EWS. The architecture is cloud-agnostic, meaning that its components can be deployed in public, private, or hybrid cloud environments without dependence on the proprietary services of a particular cloud provider.

Fig. 4 shows how regional anomaly identification, hierarchical context orchestration, CEP-based warning confirmation, and alert dissemination are connected in one cloud-agnostic monitoring architecture.

The study has several limitations. The validation was performed on synthetic data, which allows controlled inspection of anomaly types but does not fully reproduce the complexity of real climate and sensor data. The thresholds and weights of the anomaly score were selected for simulation purposes and require calibration on historical observations. In addition, the proposed architecture was evaluated conceptually and through simulation rather than through full industrial deployment. These limitations define the next stage of validation, which should include calibration on real historical climate observations, testing on streaming sensor data, sensitivity analysis of weights and thresholds, and evaluation in a pilot deployment scenario.

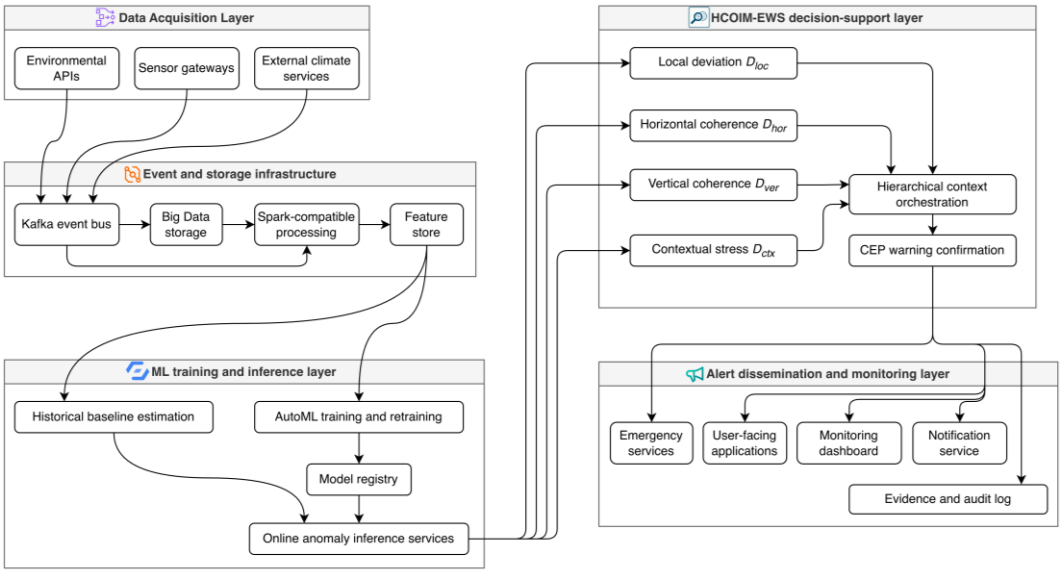

Fig. 4. Cloud-agnostic HCOIM-EWS reference architecture connecting environmental data acquisition, event streaming, ML training and inference, hierarchical context orchestration, CEP-based warning confirmation, and alert dissemination. Source: developed by the authors

CONCLUSION

This paper proposed a cloud-agnostic intelligent monitoring architecture for climate anomalies in early warning systems, supported by an integrated ML-based anomaly-identification and decision-support method. The architecture combines a formal hierarchical anomaly model, region-specific anomaly detector training, warning-oriented decision logic, the author-defined HCOIM-EWS method, and a reusable cloud-agnostic reference architecture.

The proposed anomaly model integrates local deviation, horizontal coherence, vertical coherence, and contextual stress. These components allow the system to distinguish isolated local deviations from spatially and hierarchically meaningful warning patterns. The HCOIM-EWS method operationalizes this model by transforming local anomaly candidates into warning-level evidence using persistence, neighborhood support, hierarchical consistency, and CEP-based escalation rules.

Simulation on a synthetic climate graph with 36 regions and 33 injected anomaly episodes showed that the author-defined HCOIM-EWS intelligent monitoring method achieved the lowest false-alarm count at a comparable event-recall level relative to the

Robust Z-score statistical method, the Isolation Forest anomaly-detection algorithm, and the One-Class SVM method. The proposed method reduced false alarms by 55.0% compared with the Robust Z-score statistical method, 22.6% compared with the Isolation Forest anomaly-detection algorithm, and 49.7% compared with the One-Class SVM method, while preserving comparable event-level recall.

The obtained results confirm that the proposed method is suitable for warning-oriented anomaly identification, where false-alarm reduction, contextual interpretation, and decision-support reliability are as important as point-level anomaly sensitivity. Future work should include validation on real historical and streaming climate data, calibration of thresholds and CEP rules, integration of data- and model-drift management, and extension of XAI and MLOps mechanisms for industrial-scale early warning systems.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [V.L., Y.M., P.K., A.P., T.K.]; methodology, [V.L., Y.M., T.K.]; validation, [V.L., Y.M., P.K.]; formal analysis, [V.L., Y.M., A.P.]; investigation, [Y.M., P.K., A.P.]; resources, [P.K., T.K.]; data curation, [A.P.]; writing – original draft preparation, [V.L., Y.M., T.K.]; writing – review and editing, [V.L., Y.M., T.K.]; visualization, [T.K., A.P.]; supervision, [V.L.]; project administration, [V.L.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Rokhideh, M., Fearnley, C., & Budimir, M. (2025). Multi-hazard early warning systems in the Sendai Framework for disaster risk reduction: Achievements, gaps, and future directions. *International Journal of Disaster Risk Science, 16*, 103–116. https://doi.org/10.1007/s13753-025-00622-9

[2] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys, 41*(3), Article 15. https://doi.org/10.1145/1541880.1541882

[3] Sgueglia, A., Di Sorbo, A., Visaggio, C. A., & Canfora, G. (2022). A systematic literature review of IoT time series anomaly detection solutions. *Future Generation Computer Systems, 134*, 170–186. https://doi.org/10.1016/j.future.2022.04.005

[4] Camps-Valls, G., Fernández-Torres, M.-Á., Cohrs, K.-H., Höhl, A., Castelletti, A., Pacal, A., Robin, C., Martinuzzi, F., Papoutsis, I., Prapas, I., Pérez-Aracil, J., Weigel, K., Gonzalez-Calabuig, M., Reichstein, M., Rabel, M., Giuliani, M., Mahecha, M. D., Popescu, O.-I., Pellicer-Valero, O. J., ... Williams, T. (2025). Artificial intelligence for modeling and understanding extreme weather and climate events. *Nature Communications, 16*, Article 1919. https://doi.org/10.1038/s41467-025-56573-8

[5] Spohn, T. K., Walsh, E., Horan, K., O'Donoghue, J., Charnecki, T., Haslam, M., & Gallagher, S. (2026). A machine learning approach using autoencoders to perform quality control on meteorological data. *Environmental Data Science, 5*, Article e1. https://doi.org/10.1017/eds.2026.10030

[6] Sun, H., Liu, Y., Zhang, Y., et al. (2025). A machine learning-based quality control algorithm for heavy rainfall observations. *Remote Sensing, 17*(24), Article 3976. https://doi.org/10.3390/rs17243976
[7] Islam, M. M., Hasan, M., Mia, M. S., Masud, A. A., & Islam, A. R. M. T. (2025). Early warning systems in climate risk management: Roles and implementations in eradicating barriers and overcoming challenges. *Natural Hazards Research, 5*(3), 523–538. https://doi.org/10.1016/j.nhres.2025.01.007
[8] Chovanec, D., Kollár, B., Halúsková, B., Kubás, J., Pawęska, M., & Ristvej, J. (2025). A component-based approach to early warning systems: A theoretical model. *Applied Sciences, 15*(6), Article 3218. https://doi.org/10.3390/app15063218
[9] Girotto, C. D., Piadeh, F., Bkhtiari, V., Behzadian, K., Chen, A. S., Campos, L. C., & Zolgharni, M. (2024). A critical review of digital technology innovations for early warning of water-related disease outbreaks associated with climatic hazards. *International Journal of Disaster Risk Reduction, 100*, Article 104151. https://doi.org/10.1016/j.ijdrr.2023.104151
[10] Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M. J., Ghodsi, A., Gonzalez, J., Shenker, S., & Stoica, I. (2016). Apache Spark: A unified engine for big data processing. *Communications of the ACM, 59*(11), 56–65. https://doi.org/10.1145/2934664
[11] Hutter, F., Kotthoff, L., & Vanschoren, J. (Eds.). (2019). *Automated machine learning: Methods, systems, challenges*. Springer. https://doi.org/10.1007/978-3-030-05318-5
[12] Beck, H. E., McVicar, T. R., Vergopolan, N., Berg, A., & Wood, E. F. (2023). High-resolution (1 km) Köppen–Geiger maps for 1901–2099 based on constrained CMIP6 projections. *Scientific Data, 10*, Article 724. https://doi.org/10.1038/s41597-023-02549-6
[13] García, J. A., Acero, F. J., Martínez-Pizarro, M., & Lara, M. (2024). A Bayesian hierarchical spatio-temporal model for summer extreme temperatures in Spain. *Stochastic Environmental Research and Risk Assessment, 38*(9), 3393–3410. https://doi.org/10.1007/s00477-024-02754-8
[14] Dereszynski, E. W., & Dietterich, T. G. (2011). Spatiotemporal models for data-anomaly detection in dynamic environmental monitoring campaigns. *ACM Transactions on Sensor Networks, 8*(1), Article 3. https://doi.org/10.1145/1993042.1993045
[15] Bâra, A., Văduva, A. G., & Oprea, S.-V. (2024). Anomaly detection in weather phenomena: News and numerical data-driven insights into climate change in Romania's historical regions. *International Journal of Computational Intelligence Systems, 17*, Article 134. https://doi.org/10.1007/s44196-024-00536-2
[16] Lyashkevych, V. Y. (2026). Intelligent monitoring as an information technology for context-aware decision-making strategy selection. *Ukrainian Journal of Information Technology, 8*(1), 39–50. https://doi.org/10.23939/ujit2026.01.039
[17] Lyashkevych, V. Y. (2025). Sustainable optimization of consolidated data processing algorithms based on machine learning and genetic algorithms. *Electronics and Information Technologies, 32*, 39–54. https://doi.org/10.30970/eli.32.3
[18] Gura, V., Ostrovska, O., & Lyashkevych, V. (2025). Scenario modelling and impact estimation of a local pollutant on the environment. *Applied Computer Systems, 30*(1), 195–201. https://doi.org/10.2478/acss-2025-0021

[19] Coughlan de Perez, E., Berse, K. B., Depante, L. A. C., Easton-Calabria, E., Evidente, E. P. R., Ezike, T., Heinrich, D., Jack, C., Lagmay, A. M. F. A., Lendelvo, S., Marunye, J., Maxwell, D. G., Murshed, S. B., Orach, C. G., Pinto, M., Poole, L. B., Rathod, K., Shampa, & Van Sant, C. (2022). Learning from the past in moving to the future: Invest in communication and response to weather early warnings to reduce death and damage. *Climate Risk Management, 38*, Article 100461. https://doi.org/10.1016/j.crm.2022.100461

[20] Sadiq, A.-A., Dougherty, R. B., Tyler, J., & Entress, R. (2023). Public alert and warning system literature review in the USA: Identifying research gaps and lessons for practice. Natural Hazards, 117(2), 1711–1744. https://doi.org/10.1007/s11069-023-05926-x

ХМАРНО-НЕЗАЛЕЖНА АРХІТЕКТУРА ІНТЕЛЕКТУАЛЬНОГО МОНІТОРИНГУ КЛІМАТИЧНИХ АНОМАЛІЙ У СИСТЕМАХ РАННЬОГО ПОПЕРЕДЖЕННЯ

Василь Ляшкевич*, Юрій Марчук, Петро Кулик, Андрій Патинко, Тарас Керод
Львівський національний університет імені Івана Франка, вул. Драгоманова, 50, Львів, 79005, Україна

АНОТАЦІЯ

Вступ. Кліматичні небезпеки вимагають систем раннього попередження, здатних поєднувати гетерогенні спостереження за станом довкілля, виявляти аномальні просторово-часові закономірності та передавати попереджувальні сигнали зацікавленим сторонам. Наявні рішення часто розділяють програмну архітектуру, моделі виявлення аномалій і логіку прийняття рішень, яка перетворює відхилення на ранні сигнали.

Матеріали та методи. У дослідженні розроблено хмарно-незалежну архітектуру інтелектуального моніторингу для систем раннього попередження, що підтримується ідентифікацією аномалій і прийняттям рішень на основі машинного навчання. Вона оцінює чотири типи ознак: відхилення від нормального стану, узгодженість із сусідніми регіонами, узгодженість із батьківською областю та вплив контекстних чинників. Синтетичний просторово-часовий кліматичний граф використано для порівняння статистичного методу Robust Z-score, алгоритму виявлення аномалій Isolation Forest, методу One-Class Support Vector Machine та запропонованого методу HCOIM-EWS, де HCOIM-EWS означає Hierarchical Context-Orchestrated Intelligent Monitoring for Early Warning Systems.

Результати. Запропонований метод інтегрує вибір детекторів на основі машинного навчання, регіонально-специфічне навчання, ієрархічну оцінку аномалій, логіку прийняття рішень щодо попереджень на основі системи опрацювання комплексних подій та хмарно-незалежну еталонну архітектуру моніторингу. У моделюванні HCOIM-EWS досяг найнижчої кількості хибних тривог за порівнюваного рівня повноти виявлення подій. Порівняно з Robust Z-score, Isolation Forest і One-Class SVM запропонований метод зменшив кількість хибних тривог на 55,0 %, 22,6 % і 49,7 % відповідно. Метод придатний для інтелектуального моніторингу, орієнтованого на попередження, де операційна надійність і зменшення хибних тривог є такими ж важливими, як і чутливість до точкових аномалій.

Висновки. Запропонований метод розширює виявлення аномалій у системах раннього попередження від аналізу ізольованих локальних часових рядів до ієрархічного, контекстно-залежного та подієво-орієнтованого інтелектуального моніторингу. Це підтверджує технічну доцільність архітектури та обґрунтовує її використання як багаторазового шаблону для платформ моніторингу кліматичних ризиків.

Ключові слова: виявлення аномалій, системи раннього попередження, інтелектуальний моніторинг, хмарно-незалежна архітектура, обробка складних подій, машинне навчання

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
02 June, 2026	24 June, 2026	25 June, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 83–94
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC 004.89

CONVOLUTIONAL NEURAL NETWORKS AND GRADIENT BOOSTING COMPARATIVE ANALYSIS FOR MULTI-HORIZON CLASSIFICATION OF CRYPTOCURRENCY STREAM DATA

Ivan Khamar
Department of Radioelectronic and Computer Systems,
Ivan Franko National University of Lviv,
50 Drahomanova Str., Lviv, 79005, Ukraine

Khamar, I. (2026). Convolutional Neural Networks and Gradient Boosting Comparative Analysis for Multi-Horizon Classification of Cryptocurrency Stream Data. *Electronics and Information Technologies*, 34, 83–94. https://doi.org/10.30970/eli.34.5

ABSTRACT

Introduction. Cryptocurrency price direction classification is a binary classification task for time series with low composite signal/sum. Image encoding methods (GASF, MTF, Recurrence Plot) transform one-dimensional time series into two-dimensional images for processing by CNNs. However, a systematic comparison of the performance of such approaches with classical sequential and tabular methods for different data scales and forecasting horizons is lacking.

Materials and methods. A streaming pipeline based on Apache Kafka is proposed for collecting and processing OHLCV data of cryptocurrency trading pairs from the Binance exchange. Three series of experiments of different scales are conducted: small (46 pairs, 3 months, 10 948 samples), medium (91 pairs, 6 months, 46 355) and large (147 pairs, 12 months, 139 887). For each scale, six CNN architectures are compared – five with 2D image coding (SimpleCNN, HybridCNN, HybridResCNN, EfficientNet-B0, MobileNetV2) and one 1D control (CNN1D-Hybrid) – with gradient boosting (LightGBM) on tabular features. Classification was performed for four forecast horizons (1, 3, 6, and 12 intervals of 8 hours). Statistical significance was assessed using the McNemar criterion.

Results and Discussion. At a small scale, the 1D control model achieves the highest Matthews correlation coefficient (MCC) = +0.21 at horizon h = 1, while the best 2D image-based result is only MCC = +0.11. At medium scale, all CNN models exhibit MCC ≈ 0, while LightGBM consistently dominates with MCC up to +0.09 at horizons h = 6 and h = 12. At large scale, no model exceeds MCC = +0.07, indicating saturation of the predictive signal with increasing data heterogeneity. 2D image-based coding does not provide a consistent advantage over 1D sequential and tabular approaches at any scale.

Conclusion. The transformation of time series into two-dimensional images for convolutional neural networks does not provide a sustainable advantage over tabular gradient boosting and 1D convolutional networks in classifying cryptocurrency price direction. Naive scaling by increasing the number of trading pairs and time window worsens results of all model classes due to increasing heterogeneity of market regimes. These results indicate a need for regime-adaptive and cluster approaches instead of building a single global model.

Keywords: data classification; time series imaging; convolutional neural networks; gradient boosting; data streaming; Apache Kafka.

INTRODUCTION

Predicting the direction of price movements for financial assets is one of the key challenges of algorithmic trading and automated decision-making in financial markets. The cryptocurrency market, characterized by high volatility, round-the-clock trading, and a significant amount of noise, presents a particularly challenging environment for building predictive models [1]. Unlike traditional stock markets, cryptocurrency data flows in a continuous stream from hundreds of trading pairs simultaneously, creating a need for scalable approaches to real-time processing and classification [2].

In recent years, researchers have focused considerable attention on methods of image-based encoding of time series, in particular the Gramian Angular Summation Field (GASF) and the Markov Transition Field (MTF) [3] and Recurrence Plot (RP) [4]. These methods transform a one-dimensional time series into a two-dimensional image, which is then processed by convolutional neural networks (CNNs). The theoretical advantage of this approach is the ability of CNNs to detect spatial patterns and correlations between different time points that may be invisible to sequential models [5]. Some studies demonstrate the potential of image-based coding for classifying time series in financial forecasting [6] and monitoring the cryptocurrency market [7]. However, its effectiveness in the financial domain, where the signal-to-noise ratio is significantly lower, remains a subject of debate.

On the other hand, classical machine learning methods based on tabular features, particularly gradient boosting (LightGBM, XGBoost), demonstrate consistently high performance in financial time series classification tasks [8]. These models work directly with engineered features (technical indicators, window statistics) without an intermediate transformation into images and train orders of magnitude faster than deep neural networks. At the same time, tabular methods do not utilize information about the spatial structure of the time series, which image-based encoding is potentially capable of capturing.

Several factors make it difficult to objectively compare these approaches. First, most existing studies evaluate models on a limited number of currency pairs and a fixed time horizon, which does not reflect real-world streaming conditions [9]. Second, the influence of the tabular component is rarely controlled for: hybrid models may perform well due to their tabular branch rather than graphical encoding. Third, the effect of data scale has not been systematically investigated, although more data may not improve models when trading pairs exhibit fundamentally different market regimes.

To address these issues, this paper proposes a controlled experiment involving a systematic comparison of CNN architectures based on image encoding, one-dimensional convolutional networks, and gradient boosting on tabular features. All neural models were built with a single table branch and the embedding of a trading pair identifier, which allows us to isolate the contribution of the image component specifically. The study was conducted on three data scales: small (50 trading pairs, 3 months), medium (100 pairs, 6 months), and large (200 pairs, 12 months), with four forecasting horizons (1, 3, 6, 12 intervals of 8 hours each). After data cleaning number of pairs was reduced to 46, 91 and 147 pairs. Data collection and preprocessing were implemented in a streaming pipeline based on Apache Kafka, which replicates the conditions of a real-world streaming system.

A novel aspect of this study is the multi-scale setting with controlled ablation. Unlike studies that compare models on a single fixed dataset, we systematically vary two measures of complexity – the number of trading pairs and the time window length – and investigate how these factors interact with the forecasting horizon and the type of input data encoding. This allows us not only to determine which architecture performs best under specific conditions but also to identify patterns of model degradation when scaling, which is critically important for designing real-world streaming decision-making systems.

MATERIALS AND METHODS

Data sources and the data collection pipeline

Cryptocurrency markets generate a continuous stream of quotes across hundreds of trading pairs simultaneously, which fundamentally distinguishes them from traditional stock markets with fixed trading sessions. To replicate these conditions, this work builds a streaming pipeline based on Apache Kafka, which collects OHLCV data (Open, High, Low, Close, Volume) from the Binance exchange's public API at 8-hour candle intervals. The Kafka producer periodically polls the API and publishes the received records to a topic, from which the Kafka consumer reads them for further feature extraction and the formation of training samples. This approach ensures the reproducibility of the experiment and simulates the real architecture of a streaming decision-making system [2]. Trading pairs were selected based on the criterion of the highest average daily trading volume in pairs with USDT. To systematically study the impact of data scale, the experiment was conducted in three configurations, the parameters of which are shown in Table 1. The chronological split into training, validation, and test samples (64% / 16% / 20%) was performed without shuffling to prevent information leakage from future time points into past ones.

Feature engineering and target variable

For each trading pair, a set of table-based features is calculated based on OHLCV data, covering technical indicators (moving averages, RSI, MACD, Bollinger Bands, ATR), statistical characteristics of the window (volatility, asymmetry, and excess of the return distribution), and volume metrics (the ratio of current volume to the moving average). These features are calculated on a fixed window of the latest observations and fed into the input of the tabular branch of hybrid models and the gradient boosting model. Each trading pair is also assigned an integer identifier, which is used to construct a training embedding, allowing the model to account for the individual behavioral characteristics of a specific instrument.

The target variable is defined as a binary indicator of the direction of the closing price change over a given forecasting horizon:

$$y_t = \begin{cases} 1, \text{if } C_{t+h} > C_t \\ 0, \text{if } C_{t+h} \leq C_t \end{cases} \quad (1)$$

where C_t – is the closing price at time t, and $h \in \{1,3,6,12\}$ – is the forecasting horizon, expressed in terms of the number of 8-hour intervals. Thus, the problem reduces to binary classification: the model predicts whether the price will rise relative to the current level over the next 8, 24, 48, or 96 hours, respectively.

Image encoding of time series

To transform one-dimensional time series into two-dimensional images, three encoding methods were applied, each of which captures a different aspect of the series' dynamics.

The Gramian Angular Summation Field (GASF) encodes pairwise trigonometric relationships between the normalized values of the series. The input time series $X = \{x_1, x_2, \dots, x_n\}$ is first normalized to the interval [–1, 1]. After which each value $\tilde{x}_i$ s mapped to an angular representation $\varphi_i = arccos(\tilde{x}_i)$. The GASF matrix is defined as:

$$G_{ij} = \cos(\varphi_i + \varphi_j), \quad (2)$$

where G_{ij} encodes the temporal correlation between positions i and j. The diagonal of the matrix contains information about the absolute values of the series, while the off-diagonal elements contain information about the relationships between different time points.

The Markov Transition Field (MTF) represents the transition probabilities between the discrete states of a sequence. The range of values is divided into Q quantile bins, and for each pair of bins (q_i, q_j). The transition frequency w_{ij} from bin q_i to bin q_j per step is calculated. The MTF matrix is constructed as follows:

$$M_{ij} = w_{qi \to qj}, \tag{3}$$

where q_i and q_j are the bins to which the series values at times i and j, respectively, belong. The MTF thus encodes the dynamics of transitions between price levels, allowing for the identification of sustained trends and reversal points [3].

A Recurrence Plot (RP) visualizes the recurrence of states of a dynamic system in phase space. The elements of the binary matrix are defined as:

$$R_{ij} = \theta(\varepsilon - |x_i - x_j|), \tag{4}$$

where θ is the Heaviside function, ε is the similarity threshold radius, and $|\cdot|$ is the Euclidean norm. A value of R_{ij} =1 indicates that the states at times i and j are close within the threshold ε. RP effectively detects cyclic patterns, regime changes, and laminar phases in time series [4].

The three resulting matrices are combined into a three-dimensional image as separate channels (similar to RGB), forming a 32×32×6 input for convolutional models (Fig. 1).

Image encoding was performed using a time-series window of 32 candles (window_size = 32), producing 32×32 matrices for all three methods. For MTF, the value range was discretized into Q = 8 quantile bins. For Recurrence Plot, an adaptive similarity threshold $\varepsilon = 0.5 \cdot \sigma(\tilde{x})$ was applied, where $\sigma(\tilde{x})$ is the standard deviation of the normalized series. The full implementation is available in the Kaggle notebook [10].

This approach allows CNN to simultaneously analyze angular dependencies (GASF), transition dynamics (MTF), and the recurrent structure (RP) of a time series in a single image.

Model architectures

All models under study are divided into three categories based on the type of input data: models based on 2D image encoding, a 1D control model, and tabular gradient boosting. To ensure a controlled comparison, the hybrid neural models are built using a unified two-branch architecture: the image (or sequential) branch extracts features from the time series, while the tabular branch processes engineered features along with a learned embedding of the trading pair identifier. The outputs of both branches are concatenated in a shared fusion head, which generates a single logit for binary classification.

The tabular branch is common to all hybrid models. The trade pair identifier is mapped to an 8-dimensional embedding vector, which is concatenated with the tabular features and processed by a two-layer perceptron (MLP) with a 32-dimensional output. The fusion head takes the concatenation of the image and table feature vectors, passes it through a 64-dimensional hidden layer with a ReLU activation function and 0.3 dropout, and returns a scalar logit. This unified design allows us to isolate the contribution of the image component, since the only difference between the models is the method of processing the time series.

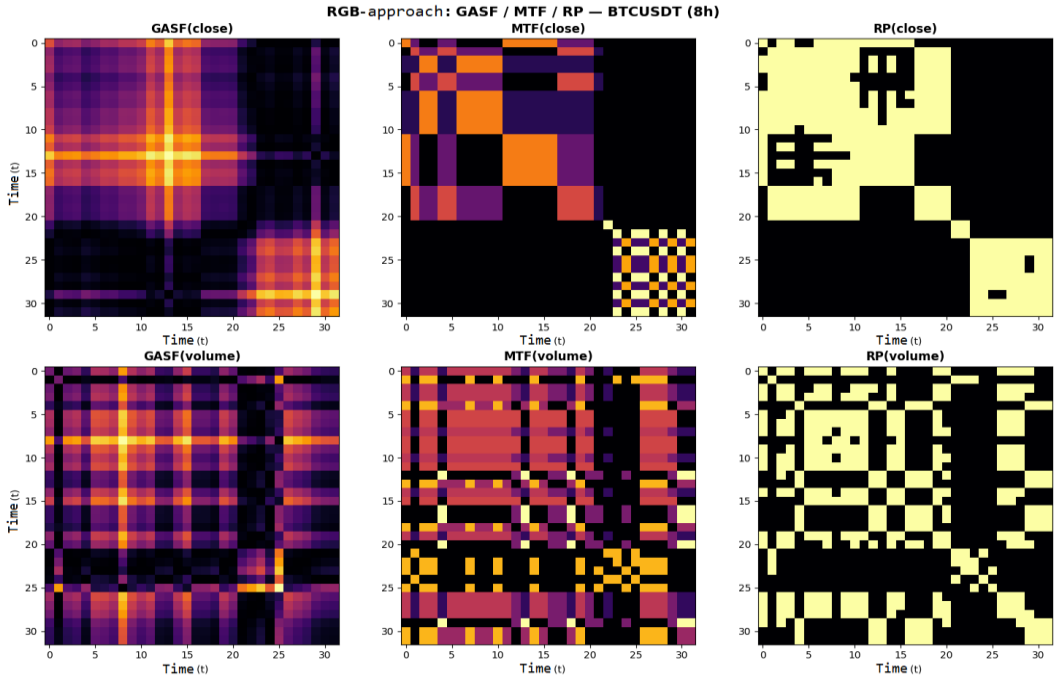

Fig. 1. RGB-approach of GASF, MTF and RP.

In the category of 2D models, five architectures of increasing complexity were investigated. SimpleCNN is the only model without a tabular branch, serving as a baseline for evaluating the effectiveness of image encoding itself. Its backbone consists of two convolutional layers (32 and 64 filters) with batch normalization and global average pooling, yielding a 128-dimensional feature vector. HybridCNN uses an identical backbone but is augmented with a tabular branch via a merge head. HybridResCNN replaces the backbone with an architecture featuring two residual blocks in the style of ResNet [11], which allows for training a deeper network without gradient degradation. The two models are built on backbones pre-trained on ImageNet: EfficientNet-B0-Hybrid [12] and MobileNetV2-Hybrid [13]. Since the input images are 32×32 with 6 channels (two time series × three encoding methods), a 1×1 adaptive convolutional layer is used to project from 6 channels to 3, after which the image is upscaled to 96×96 for compatibility with pretrained models. The last 30 parameters of the backbone are unfrozen for fine-tuning; the rest remain fixed.

CNN1D-Hybrid serves as the baseline model for testing the hypothesis regarding the advantage of 2D encoding. It receives the same raw time series (close and volume) used to construct images, but processes them directly using a one-dimensional convolutional network (three Conv1d layers with 32, 64, and 128 filters). The tabular branch and merge head are identical to those in other hybrid models. Thus, if the 2D models outperform the CNN1D-Hybrid, this can be attributed specifically to the contribution of image encoding, rather than tabular features or pair embedding.

LightGBM – a gradient boosting implementation that operates exclusively on engineered tabular features without any time-series encoding – is used as the tabular baseline. LightGBM was chosen for its consistently high performance on tabular data and its learning speed, which is several orders of magnitude faster than neural networks..

Training and assessment

All neural models are trained using the BCEWithLogitsLoss loss function and the Adam optimizer. To prevent overtraining, an early stopping mechanism with a patience of

7 epochs is applied: training is stopped if the Matthews correlation coefficient (MCC) value on the validation sample does not improve for 7 consecutive epochs, and the weights with the best validation quality are restored. The maximum number of epochs is limited to 30. The optimal probability binarization threshold is selected on the validation sample by iterating over values from 0,2 to 0,8 with a step of 0,01 according to the MCC maximization criterion. LightGBM is trained with default parameters without additional hyperparameter tuning.

This creates asymmetry in comparison conditions: while neural models benefit from early stopping and threshold tuning, LightGBM uses default parameters. This asymmetry should be considered when interpreting the advantage of the tabular approach - the gap may narrow with hyperparameter tuning of LightGBM, which is left for future work.

MCC is chosen as the main evaluation metric, which is resistant to class imbalance and takes into account all four elements of the confusion matrix, unlike accuracy and F1-score [14]. Additionally, AUC-ROC, F1-score, and Brier score are recorded. The statistical significance of the differences between pairs of models is assessed using the McNemar criterion, which compares the error distributions of two classifiers on the same test sample [15]. The parameters of the three scales of the experiment are given in **Table 1**.

Table 1. **Experimental scale parameters.**

Parameter	Small	Medium	Large
Number of trading pairs	46	91	147
Time window	3 months.	6 months.	12 months.
Training sample	7 084	30 121	90 860
Validation sample	1 610	6 944	20 848
Test sample	2 254	9 290	28 179
Forecasting horizons	1, 3, 6, 12 × 8 h		
Image size	32 × 32 × 6		
Maximum epochs / patience	30 / 7		

Execution environment

The experiments were implemented in Python 3.10 using the PyTorch 2.1 (neural networks), LightGBM (gradient boosting), and pyts (image encoding of time series) libraries. The computations were performed in the Kaggle Notebooks cloud environment with an NVIDIA Tesla P100 hardware accelerator (16 GB VRAM) and approximately 30 GB of RAM. A streaming data collection pipeline based on Apache Kafka was deployed locally. It should be noted that the training times reported in the results reflect the conditions of a cloud environment with shared resources and are not directly transferable to dedicated hardware; they are provided solely for relative comparison between models within a single execution environment.

RESULTS AND DISCUSSION

Models and Baselines

To evaluate the effectiveness of image coding, five 2D-CNN architectures (SimpleCNN, HybridCNN, HybridResCNN, EffNetB0-Hybrid, MobileNetV2-Hybrid) were compared with two alternative approaches:

1. CNN1D-Hybrid – a control model that uses the same tabular branch and merge head as the 2D hybrids, but processes raw time series using one-dimensional convolution without image encoding. Comparison with this model isolates the contribution of the 2D transformation specifically.
2. LGBM-Tabular – gradient boosting (LightGBM) that operates exclusively on engineered tabular features. It serves as the top baseline for the tabular approach.

Additionally, three naive baselines were used: Majority (always predicts the dominant class), Random (random classifier), and Persistence (predicts the continuation of the current trend). The comparison was conducted on three scales (Small, Medium, Large) for four forecasting horizons (h = 1, 3, 6, 12 intervals of 8 hours each).

Comparison of performance by scale

Table 2 presents the MCC values for the best 2D-CNN, the baseline CNN1D-Hybrid, and LGBM-Tabular at each scale and horizon.

Table 2. **MCC of the best models at each scale and horizon.**

Scale	Model	h=1	h=3	h=6	h=12
Small	Best 2D CNN	+0.115	+0.059	+0.095	+0.074
	CNN1D Hybrid	+0.212	+0.070	+0.026	-0.014
	LGBM	-0.038	-0.008	+0.095	+0.139
Medium	Best 2D CNN	+0.005	+0.035	-0.002	+0.019
	CNN1D Hybrid	-0.011	+0.055	-0.147	-0.087
	LGBM	+0.031	+0.030	+0.091	+0.093
Large	Best 2D CNN	+0.007	+0.008	+0.015	+0.033
	CNN1D Hybrid	-0.003	+0.039	+0.047	+0.017
	LGBM	+0.061	+0.022	+0.075	-0.042

At the small scale, the clearest differentiation between models is observed. CNN1D-Hybrid achieves an MCC of +0.212 at h = 1, which is the highest value in the entire study. This indicates that, on a limited sample, one-dimensional convolution is more effective at extracting short-term patterns than 2D image encoding. At the same time, on longer horizons (h = 6, h = 12), LGBM-Tabular dominates with an MCC of up to +0.139, while all CNN models degrade.

At the medium scale, there is a sharp decline in the performance of all neural models. The best 2D-CNN achieves only MCC = +0.035, and CNN1D-Hybrid exhibits negative MCC values at h = 6 and h = 12. The only model that consistently outperforms the baselines remains LGBM-Tabular, with MCC ranging from +0.030 to +0.093 across all horizons.

At large scale, no model exceeds MCC = +0.075. All 2D-CNNs exhibit MCC < +0.033, which is statistically indistinguishable from a random classifier. Notably, even LGBM-Tabular, which consistently dominated at the medium scale, shows a negative MCC of −0.042 at h = 12, indicating a general saturation of the predictive signal as data heterogeneity increases.

Ablation analysis

Table 3 shows the pairwise comparisons performed to isolate the contribution of 2D encoding. The comparisons were performed using the McNemar criterion between the best 2D-CNN and CNN1D-Hybrid. Since both models have an identical tabular branch, the

statistically significant difference in accuracy can only be explained by the effect of image encoding.

The results reveal a clear pattern: 2D encoding has an advantage mainly at longer horizons (h = 6, h = 12), where spatial patterns in the GASF/MTF/RP images reflect slower market trends. In contrast, at the short horizon (h = 1), 1D convolution consistently outperforms 2D at small and medium scales. However, this advantage of 2D at longer horizons disappears at large scale, where 2D instead shows an advantage only at the shortest horizon (h = 1). Heterogeneity of 147 pairs, which blurs the structural patterns in the images, could be the reason.

Table 3. **Ablation of the best 2D-CNN vs CNN1D-Hybrid with McNemar's test.**

Scale	h	Best 2D	Acc 2D, %	Acc 1D, %	Δ	p	Result
Small	1	HybResCNN	49,0	60,9	−11,9	<0,001	2D worse
	6	HybCNN	65,1	61,7	+3,5	0,012	2D better
	12	MobNetV2	71,8	58,9	+12,9	<0,001	2D better
Medium	1	HybCNN	45,8	50,3	−4,5	<0,001	2D worse
	6	EffB0	51,6	44,1	+7,5	<0,001	2D better
	12	HybResCNN	51,7	43,0	+8,7	<0,001	2D better
Large	1	HybResCNN	54,0	50,8	+3,2	<0,001	2D better
	6	EffB0	47,37	57,1	−9,73	<0,001	2D worse
	12	HybResCNN	59,76	60,41	−0,65	0,071	No significant difference

Discussion

The results demonstrate that image-based encoding of time series (GASF, MTF, RP) does not provide a universal improvement for financial data classification tasks. Although some studies report promise for financial forecasting [6] and the cryptocurrency market [7]. Our systematic multi-scale comparison demonstrates that this advantage is not replicated. The key difference lies in the nature of the signal: in physiological or industrial time series, there are stable spatial patterns (e.g., the morphology of the QRS complex) that CNNs can effectively recognize in a 2D representation. In cryptocurrency data with a low signal-to-noise ratio, such stable visual structures are either absent or too weak to compensate for the additional complexity of 2D encoding [16]. This explains why simpler 1D and tabular approaches prove competitive or superior – they avoid a transformation that adds no information but introduces noise from discretization and compression to 32×32.

The practical implication of these results concerns the design of real-world streaming decision-making systems. Training a single 2D-CNN model at scale takes nearly 15 minutes, whereas LightGBM takes less than a minute. Given periodic retraining across four horizons, this computational gap becomes critical – if 2D encoding provides no sustained quality gain, its use in streaming systems wastes resources. Available hyperparameter tuning should be taken into account as a limitation of current research and an improvement possibility for future work.

At the same time, on a small scale with longer horizons (h = 6, h = 12), 2D models demonstrate a statistically significant advantage, leaving room for their targeted application in highly specialized scenarios with a limited number of currency pairs.

A significant limitation of the study is the use of a single global model for all currency pairs. The increase in heterogeneity as the scale expands – from 46 to 147 pairs with

fundamentally different market regimes – is a likely cause of the degradation of all models at large scales. Training-based pair embedding only partially mitigates this problem, as it encodes only the identity of the pair, not its current market regime. Clustered or regime-adaptive approaches could mitigate this effect but are beyond the current scope. Furthermore, the fixed image size of 32×32 may be insufficient to preserve fine-grained structures in GASF/MTF/RP – increasing the resolution to 64×64 or 128×128 would potentially improve the performance of 2D models at the cost of a proportional increase in training time.

Despite the generally low absolute MCC values, the study reveals a non-trivial interaction structure between encoding type, horizon, and scale, which has practical value. The very fact that data scaling degrades the results of all classes of models is an important negative result: it refutes the intuitive assumption that “more data means a better model,” which is often taken as an axiom, and points to the need for qualitative data selection instead of quantitative data augmentation.

CONCLUSION

This paper presents a systematic comparison of 2D image-based time series encoding methods (GASF, MTF, Recurrence Plot) with one-dimensional convolutional networks and tabular gradient boosting for the task of classifying cryptocurrency price trends. The central contribution is a multiscale framework with controlled ablation: a unified two-branch architecture allowed us to isolate the influence of the image component specifically, while three data scales revealed patterns that were not apparent in a single fixed dataset.

Experiments across three scales (46,91,147 trading pairs, 3–12 months) and four forecasting horizons showed that 2D encoding does not provide a consistent advantage. At the small scale, CNN1D-Hybrid achieved the highest MCC = +0.212 at $h = 1$, outperforming all 2D models. At the medium scale, all neural models were outperformed by LightGBM (MCC up to +0.093). At a large scale, no model exceeded MCC = +0.075, indicating saturation of the predictive signal as data heterogeneity increases. The McNemar’s criterion confirmed that 2D encoding significantly outperforms 1D in only 5 out of 12 scale × horizon combinations.

From a practical standpoint, these results imply that for streaming classification systems of cryptocurrency data, the use of image-based encoding with convolutional networks does not justify the computational cost compared to tabular gradient boosting, which trains several orders of magnitude faster and demonstrates more stable performance over medium forecasting horizons.

Promising directions include cluster-based or regime-adaptive models to account for heterogeneous market regimes; higher image resolution (64×64, 128×128) to preserve finer structures; and attention-based architectures (Transformer) for weighting contributions of different pairs and intervals.

In summary, the study demonstrates the importance of controlled multiscale evaluation when selecting an architecture for financial time series. The intuitive assumption that more complex data representations and larger sample sizes are superior is not supported in domains with a low signal-to-noise ratio. This conclusion is transferable to other tasks of classifying noisy time series, where the choice between image-based encoding and tabular methods requires empirical verification rather than a priori assumptions.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The author received no financial support for the research, writing, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The author declares that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

The author has read and agreed to the published version of the manuscript.

REFERENCES

[1] Fang, F., Ventre, C., Basios, M., Kanthan, L., Martinez-Rego, D., Wu, F., & Li, L. (2022). Cryptocurrency trading: A comprehensive survey. Financial Innovation, 8(1), 13. https://doi.org/10.1186/s40854-021-00321-6

[2] Kreps, J., Narkhede, N., & Rao, J. (2011). Kafka: A distributed messaging system for log processing. Proceedings of the NetDB Workshop, 1–7.

[3] Wang, Z., & Oates, T. (2015). Imaging time-series to improve classification and imputation. Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI), 3939–3945. https://doi.org/10.48550/arXiv.1506.00327

[4] Eckmann, J.-P., Kamphorst, S. O., & Ruelle, D. (1987). Recurrence plots of dynamical systems. Europhysics Letters, 4(9), 973–977. https://doi.org/10.1209/0295-5075/4/9/004

[5] Hatami, N., Gavet, Y., & Debayle, J. (2018). Classification of time-series images using deep convolutional neural networks. Proceedings of the 10th International Conference on Machine Vision (ICMV), 10696, 106960Y. https://doi.org/10.1117/12.2309486

[6] Barra, S., Carta, S. M., Corriga, A., Podda, A. S., & Recupero, D. R. (2020). Deep learning and time series-to-image encoding for financial forecasting. IEEE/CAA Journal of Automatica Sinica, 7(3), 683–692. https://doi.org/10.1109/JAS.2020.1003132

[7] Chen, W., Xu, H., Jia, L., & Gao, Y. (2021). Machine learning model for Bitcoin exchange rate prediction using economic and technology determinants. International Journal of Forecasting, 37(1), 28–43. https://doi.org/10.1016/j.ijforecast.2020.02.008

[8] Jaquart, P., Dann, D., & Weinhardt, C. (2021). Short-term bitcoin market prediction via machine learning. Journal of Finance and Data Science, 7, 45–66. https://doi.org/10.1016/j.jfds.2021.03.001

[9] Qureshi, S. M., Saeed, A., Ahmad, F., Khattak, A. R., Almotiri, S. H., Al Ghamdi, M. A., & Rukh, M. S. (2025). Evaluating machine learning models for predictive accuracy in cryptocurrency price forecasting. *PeerJ Computer Science*, 11, e2626. https://doi.org/10.7717/peerj-cs.2626

[10] Khamar, I. (2026). CNN and gradient boosting comparative analysis for multi-horizon classification of cryptocurrency stream data [Kaggle Notebook]. Kaggle. https://www.kaggle.com/code/woodborder/v6-compvision-classification

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778. https://doi.org/10.1109/CVPR.2016.90

[12] Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 6105–6114. https://doi.org/10.48550/arXiv.1905.11946

[13] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE*

Conference on Computer Vision and Pattern Recognition (CVPR), 4510–4520. https://doi.org/10.1109/CVPR.2018.00474
[14] Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics, 21(1), 6. https://doi.org/10.1186/s12864-019-6413-7
[15] McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika, 12(2), 153–157. https://doi.org/10.1007/BF02295996
[16] Bouteska, A., Abedin, M. Z., Hajek, P., & Yuan, K. (2024). Cryptocurrency price forecasting – A comparative analysis of ensemble learning and deep learning methods. International Review of Financial Analysis, 92, 103055. https://doi.org/10.1016/j.irfa.2023.103055

ПОРІВНЯЛЬНИЙ АНАЛІЗ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ ТА ГРАДІЄНТНОГО ПІДСИЛЕННЯ ДЛЯ БАГАТОГОРИЗОНТНОЇ КЛАСИФІКАЦІЇ КРИПТОВАЛЮТНИХ ПОТОКОВИХ ДАНИХ

Іван Хамар
Кафедра радіоелектронних та комп'ютерних систем,
Львівський національний університет імені Івана Франка,
вул. Драгоманова, 50, Львів, 79005, Україна

АНОТАЦІЯ

Вступ. Класифікація напрямку ціни криптовалюти – це завдання бінарної класифікації для часових рядів з низьким співвідношенням сигнал/шум. В останні роки широкого поширення набули методи кодування зображень, які перетворюють одновимірні часові ряди на двовимірні зображення для обробки згорткові нейронні мережі (ЗНМ). Однак систематичне порівняння продуктивності таких підходів з класичними послідовними та табличними методами для різних масштабів даних та горизонтів прогнозування відсутнє. Це включає вибір архітектури для реальних потокових систем, де обчислювальні ресурси обмежені, а обсяг та неоднорідність даних постійно зростають.

Матеріали та методи. Запропоновано потоковий конвеєр на базі Апачі Кафка для збору та обробки даних цін та обсягів торгових пар криптовалют з біржі Binance. Проведено три серії експериментів різного масштабу: малий (46 пар, 3 місяці, 10 948 вибірок), середній (91 пар, 6 місяців, 46 355) та великий (147 пар, 12 місяців, 139 887). Для кожного масштабу порівнюються шість архітектур ЗНМ – п'ять з 2D-кодуванням зображень (SimpleCNN, HybridCNN, HybridResCNN, EfficientNet-B0, MobileNetV2) та одна 1D-контрольна (CNN1D-Hybrid) – з градієнтним підсиленням (LightGBM) на табличних ознаках. Усі гібридні моделі використовують спільну табличну гілку з вбудовуванням торгових пар, що забезпечує контрольоване зменшення впливу кодування зображень. Класифікацію проводили для чотирьох прогнозних горизонтів (1, 3, 6 та 12 інтервалів по 8 годин). Статистичну значущість відмінностей оцінювали за критерієм МакНемара.

Результати. У малому масштабі 1D модель керування досягає найвищого значення коефіцієнту кореляції Метьюза (ККМ) = +0,21 на горизонті $h = 1$, тоді як найкращий результат на основі 2D зображень становить лише ККМ = +0.11. У середньому масштабі всі моделі ЗНМ демонструють ККМ ≈ 0, тоді як LightGBM

стабільно домінує з ККМ до +0,09 на горизонтах h = 6 та h = 12. У великому масштабі жодна модель не перевищує MCC = +0.07, що вказує на насичення прогнозного сигналу зі збільшенням неоднорідності даних. Кодування на основі 2D зображень не забезпечує стабільної переваги над 1D послідовним та табличним підходами в будь-якому масштабі.

Висновки. Виявлено, що наївне масштабування даних шляхом збільшення кількості торгових пар та часового вікна погіршує результати всіх класів моделей через зростання неоднорідності ринкових режимів. Перетворення часових рядів у двовимірні зображення для згорткових нейронних мереж не забезпечує стійкої переваги над табличним градієнтним підсиленням та одновимірними згортковими мережами в завданні класифікації напрямку цін криптовалют. Ці результати вказують на необхідність режимно-адаптивного та кластерного підходів замість побудови єдиної глобальної моделі.

Ключові слова: класифікація часових рядів; візуалізація часових рядів; згорткові нейронні мережі; градієнтне підсилення; криптовалюти; потокові дані; Апачі Кафка.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
06 May, 2026	30 June, 2026	30 June, 2026	?? June, 2026

Electronics and Information Technologies, 2026, 34, 95–112
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.93

EVOLUTION OF IMAGE SEGMENTATION METHODS: CLASSICAL ALGORITHMS AND DEEP LEARNING

Viktor Vdovychenko
Ivan Franko National University of Lviv,
50 Drahomanova St., 79005 Lviv, Ukraine

Vdovychenko V. M. (2026). Evolution of Image Segmentation Methods: Classical Algorithms and Deep Learning. *Electronics and Information Technologies*, 33, 95–112. https://doi.org/10.30970/eli.34.6.

ABSTRACT

Background. Text image binarization is an important preprocessing task in document analysis, optical character recognition, and automated information extraction. The task becomes challenging when images contain blur, noise, low contrast, or background degradation.

Materials and Methods. This study compares classical and lightweight neural methods for binarizing degraded text images. The evaluation was conducted on a paired text binarization dataset comprising degraded input images and their corresponding clean binary target masks. The evaluated methods included simple global thresholding, Otsu binarization, adaptive thresholding, K-means clustering, Fuzzy C-means clustering, watershed segmentation, region growing, region splitting and merging, Canny edge detection, Laplacian of Gaussian filtering, and a U-Net model. Tunable parameters were selected on the validation subset, while final results were reported on the independent test subset. The methods were assessed using Intersection over Union (IoU, Jaccard index), Dice similarity coefficient (DSC), Pixel Accuracy (PA), Precision, Recall, and execution time.

Results and Discussion. Adaptive thresholding achieved the best performance among classical methods: IoU = 0.785, DSC = 0.877, PA = 0.961, Precision = 0.851, and Recall = 0.923. Simple global thresholding was the fastest method, with a runtime of 0.0015 ms/image, but lower Precision. K-means, Fuzzy C-means, Otsu, and region growing demonstrated high Recall values above 0.97, but lower Precision, indicating that degraded background regions were often classified as foreground. Edge-based methods produced lower IoU and DSC because they mainly detected character contours. The Tiny U-Net model achieved the highest overall quality: IoU = 0.959, DSC = 0.979, PA = 0.994, Precision = 0.979, and Recall = 0.979.

Conclusion. The results show that different binarization methods are suitable for different scenarios, depending on the required accuracy, runtime constraints, and available resources. Classical methods remain useful for low-latency processing, while lightweight neural models provide higher accuracy when sufficient computational resources are available. Pixel Accuracy should be interpreted carefully because background pixels dominate document images.

Keywords: text binarization, image segmentation, thresholding, clustering, U-Net, evaluation metrics.

INTRODUCTION

Image segmentation is one of the fundamental tasks in computer vision and digital image processing. It aims to divide an image into meaningful regions or classes and is

widely used as a preprocessing or decision-support stage in many computer vision systems. The quality of segmentation directly affects the reliability of subsequent tasks such as object recognition, measurement, classification, and visual analysis. Despite significant progress in this field, accurate and computationally efficient segmentation remains challenging, especially under conditions of noise, blur, uneven illumination, and low contrast.

One practically important segmentation problem is text image binarization. In this task, text image is transformed into a binary representation, where text pixels are separated from the background. Text binarization is an essential preprocessing step for optical character recognition, document analysis, historical document restoration, and automated information extraction. Although the task may appear simple in high-contrast images, its complexity increases significantly when the input contains degradation artifacts such as blur, background noise, low contrast, or non-uniform illumination [1-2].

Over the decades, many classical methods have been proposed for image segmentation and binarization. Threshold-based approaches, such as simple global thresholding [3], Otsu binarization [4], and adaptive thresholding [5], remain widely used because of their simplicity and low computational cost. Clustering-based methods, including K-means [6] and Fuzzy C-means [7], separate pixels according to intensity similarity and can be applied without supervised training. Region-based methods, such as watershed segmentation [8], region growing [9], and region splitting and merging [10], use spatial or structural information to form image regions. Edge-based methods, including Canny edge detection [11] and Laplacian of Gaussian filtering [12], focus on detecting intensity transitions and object boundaries.

Each of these methods has its own advantages and limitations. Global thresholding methods are fast but sensitive to illumination and contrast variation. Adaptive thresholding can handle local intensity changes but strongly depends on the choice of block size and threshold offset. Clustering-based methods may detect most text pixels but can also assign noisy background regions to the foreground when intensity distributions overlap. Region-based methods can incorporate spatial structure but are sensitive to seed selection, region homogeneity criteria, or marker generation. Edge-based methods are effective for detecting character contours but do not naturally produce filled foreground masks. Therefore, the choice of a segmentation method depends not only on accuracy but also on robustness, parameter sensitivity, and computational cost.

Recent deep learning methods have significantly improved segmentation performance in many computer vision tasks and have become a dominant direction in modern segmentation research [13]. However, neural models require training data, computational resources, and careful validation. For small text images, lightweight architectures may be more appropriate than deep backbone-based models because excessive downsampling can remove fine character details. Therefore, compact encoder-decoder architectures such as lightweight U-Net [14] models are of particular interest for text binarization.

A common limitation of comparative studies is that segmentation methods are often evaluated under limited experimental conditions or on a small number of examples. In such cases, conclusions about the superiority of a method may be strongly influenced by image characteristics, parameter choices, or class imbalance. In document binarization, this is especially important because background pixels usually dominate the image area, making Pixel Accuracy (PA) [15] less informative than foreground-oriented metrics such as IoU [16], Dice similarity coefficient (DSC) [17], Precision, and Recall.

This study presents a comparative experimental evaluation of classical segmentation algorithms and a lightweight neural baseline for text image binarization. The evaluated methods include simple global thresholding, Otsu binarization, adaptive thresholding, K-means clustering, Fuzzy C-means clustering, watershed segmentation, region growing, region splitting and merging, Canny edge detection, Laplacian of Gaussian filtering, and a compact U-Net model. The experiments are conducted on a paired text binarization dataset

containing degraded input images and corresponding clean binary target masks. To ensure a fair comparison, tunable parameters are selected using a validation subset, while final results are reported on an independent test subset.

The main contribution of this study is a unified benchmark of classical and lightweight neural methods for degraded text binarization. The comparison considers not only segmentation accuracy but also parameter sensitivity, runtime efficiency, and differences between metrics. This allows a more detailed analysis of why methods with similar Pixel Accuracy may differ substantially in foreground segmentation quality and practical applicability.

MATERIALS AND METHODS

This study evaluates classical and lightweight neural methods for text image binarization, formulated as a binary segmentation task where text pixels represent the foreground class and background pixels represent the background class. The experimental protocol includes dataset preparation, validation-based parameter selection, final evaluation on an independent test subset, and runtime analysis.

Dataset

The experiments were conducted using the Text Binarization dataset available on Kaggle [18]. The dataset contains 20,000 paired examples of randomly generated multiline text images, where each degraded input image is matched with a clean binarized target mask. The input images contain blur and noise, making the dataset suitable for evaluating binarization methods under degraded image conditions. An example of the input images is shown in **Fig. 1**.

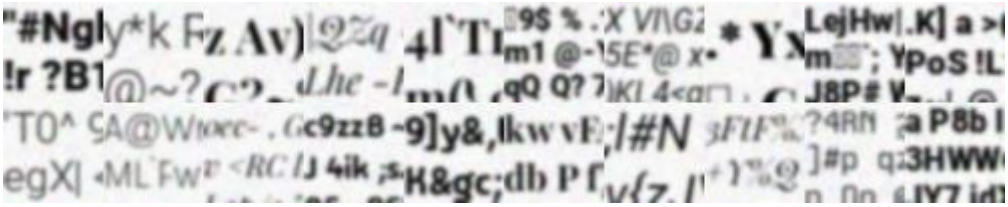

Fig. 1. Example of the dataset images

All images were processed in grayscale, and the provided binary target images were used as ground-truth masks. The dataset was divided into training, validation, and test subsets using a 70/15/15 split. The training subset was used only for the neural network model, while the validation subset was used for parameter selection in classical methods and threshold selection for the neural baseline. Final results were reported only on the independent test subset to avoid test-set-based parameter tuning.

Evaluated Methods

The study evaluated classical and lightweight neural approaches for degraded text image binarization. The classical methods included simple global thresholding, Otsu binarization, adaptive thresholding, K-means clustering, Fuzzy C-means clustering, watershed segmentation, region growing, region splitting and merging, Canny edge detection, and Laplacian of Gaussian filtering. These methods cover threshold-based, clustering-based, region-based, and edge-based segmentation strategies. [3-12]

A compact two-level U-Net [14] was additionally trained as a neural baseline. This architecture was selected because the input images have a small spatial resolution of 50×50 pixels, where deeper backbone-based models may lose fine text details due to excessive downsampling. The model consists of two encoder blocks, a bottleneck, and two decoder blocks with skip connections, producing a single-channel probability map. The

architecture of the model is shown in **Fig. 2**. It was trained using a combined binary cross-entropy and Dice loss, and the best checkpoint was selected according to validation IoU.

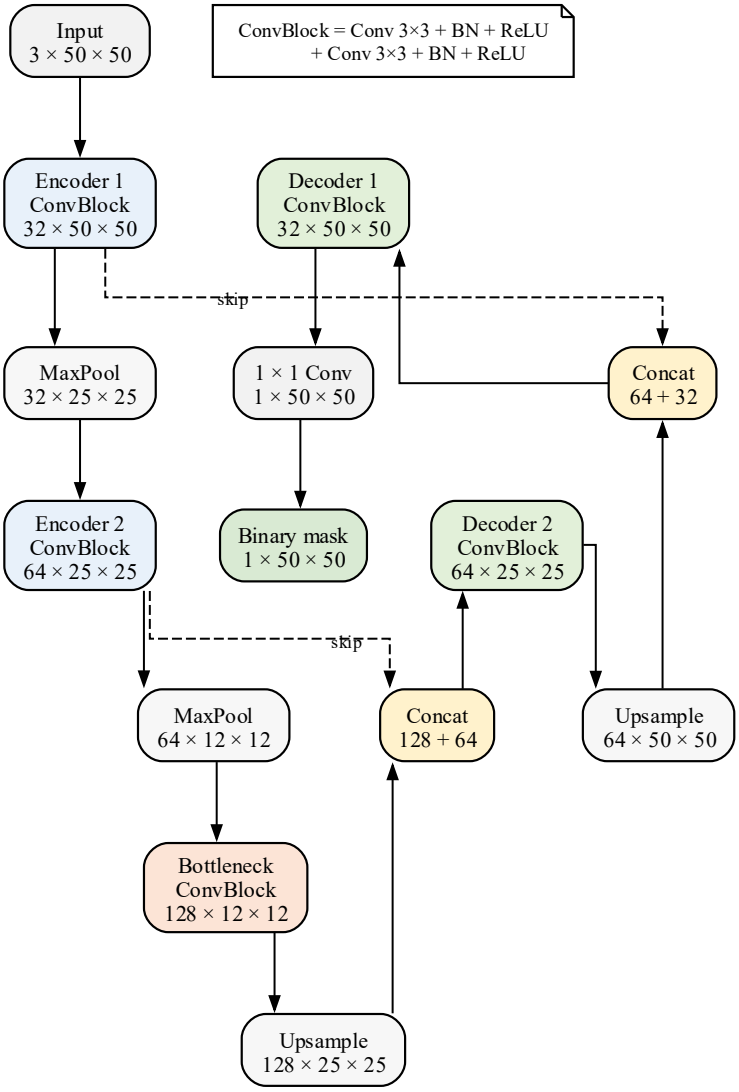

Fig. 2. Visualization of model architecture

Parameter Selection Protocol

All tunable parameters were selected on the validation subset using a predefined parameter grid. For each method, the configuration with the highest mean IoU was selected and then fixed for final evaluation on the independent test subset. Otsu binarization was the only exception, since its threshold was estimated automatically for each image. For the neural baseline, the probability threshold used to convert the output probability map into a binary mask was also selected on the validation subset.

Evaluation Metrics

IoU and Dice were used as the main foreground segmentation metrics because they directly measure the overlap between predicted and ground-truth text regions. Precision and Recall were used to analyze over-segmentation and under-segmentation behavior.

Pixel Accuracy was reported as an auxiliary metric, since background pixels dominate document images and may increase this metric even when the foreground mask is not accurately segmented.

Runtime Measurement

Computational efficiency was evaluated as execution time in milliseconds per image. For classical algorithms, runtime included only image processing time and excluded file loading and result saving. For the neural network, inference time included the forward pass of the model. Final runtime values are reported as mean milliseconds per image.

Experimental Environment

All experiments were implemented in Python 3.9. Classical methods were implemented using OpenCV and NumPy, while the neural network model was implemented using PyTorch. The experiments were conducted on a workstation equipped with an Apple M1 CPU and 8 GB of RAM, running macOS 26.4.1.

RESULTS AND DISCUSSION

Simple Global Thresholding

The global threshold value was selected on the validation subset, with the best result obtained at T = 165. As shown in **Fig. 3**, increasing the threshold improved text coverage up to an optimal range, after which additional background pixels started to be classified as foreground. On the test subset, the method achieved IoU = 0.661, DSC = 0.784, PA = 0.933, Precision = 0.751, and Recall = 0.861. The high Recall indicates that most text pixels were detected, while the lower Precision shows that some background degradation was also included in the foreground mask. The visual examples in **Fig. 4** and **Fig. 5** confirm this behavior. The method was extremely fast, with an average runtime of 0.0015 ms/image, but its robustness is limited by the use of a single global threshold.

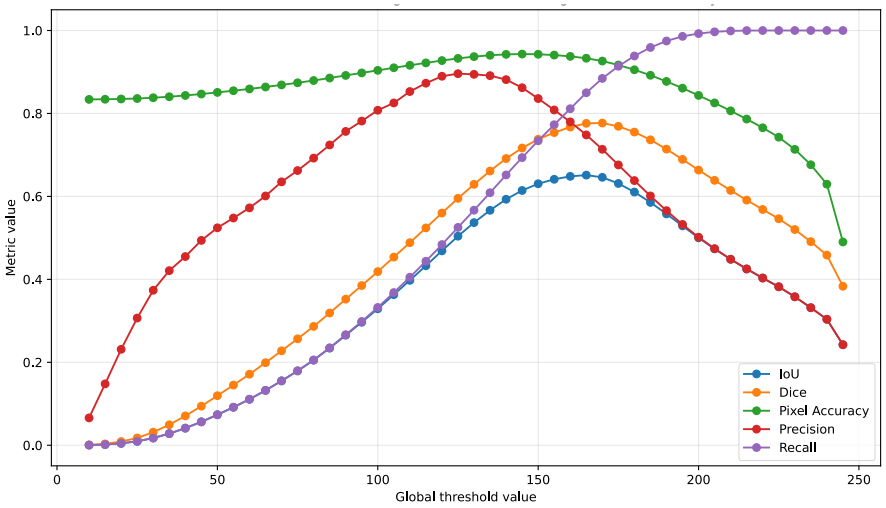

Fig. 3. Effect of the global threshold value on validation metrics for simple thresholding

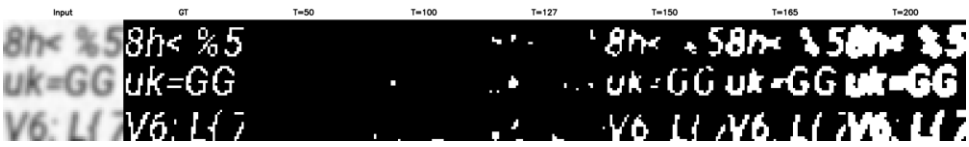

Fig. 4. Visual comparison of simple thresholding results with different threshold values on a test image

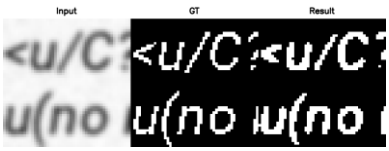

Fig. 5. Example of the best simple thresholding result on the test subset using the validation-selected threshold T=165

K-means Clustering

For K-means clustering, the number of clusters was fixed to K = 2, and the maximum number of iterations was selected on the validation subset. The best validation result was obtained with max_iter = 5, although the validation curves in **Fig. 6** show that increasing this value had almost no effect on the metrics, indicating early convergence. On the test subset, K-means achieved IoU = 0.587, DSC = 0.730, PA = 0.886, Precision = 0.596, and Recall = 0.984. The very high Recall shows that the method detected almost all text pixels, while the lower Precision indicates that many degraded background pixels were also assigned to the foreground cluster. The visual examples in **Fig. 7** and **Fig. 8** confirm this tendency toward over-segmentation. The average runtime was 0.305 ms/image, which is higher than simple thresholding but still computationally efficient.

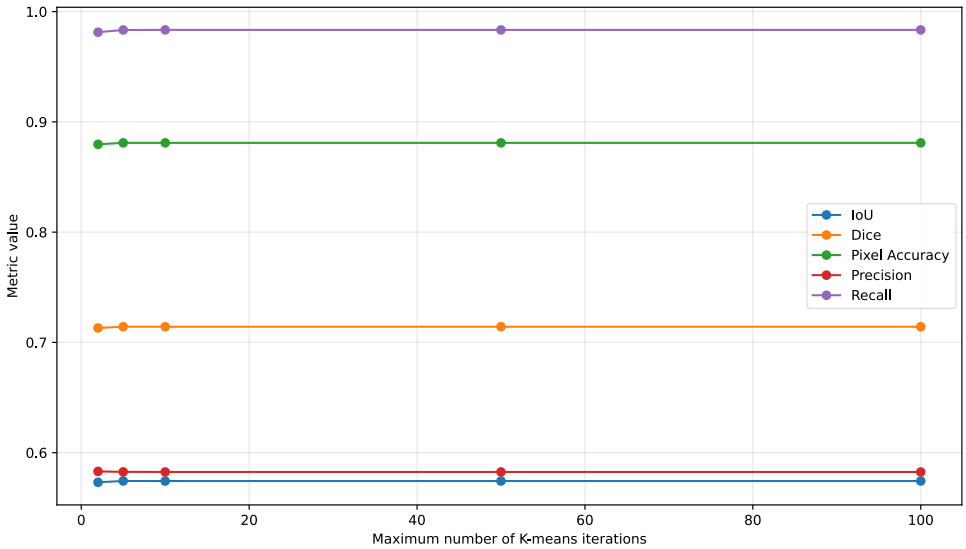

Fig. 6. Effect of the maximum number of K-means iterations on validation metrics

Fig. 7. Visual comparison of K-means results with different max_iter values on a test image

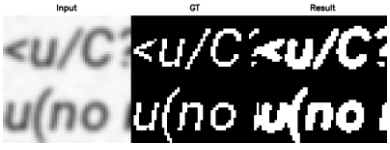

Fig. 8. Example of the best K-means result on the test subset using the validation-selected parameter max_iter = 5

Otsu Binarization

Otsu binarization was applied as an automatic thresholding method, so no validation-based parameter selection was required. On the test subset, it achieved IoU = 0.587, DSC = 0.725, PA = 0.884, Precision = 0.595, and Recall = 0.984. The very high Recall shows that Otsu detected almost all text pixels, but the lower Precision indicates that background noise and degraded regions were often included as foreground. As shown in **Fig. 9**, the automatically selected thresholds varied across images, with an average threshold of 181.853. A representative qualitative result of Otsu binarization is shown in **Fig. 10**. The average runtime was 0.0073 ms/image, confirming that Otsu remains a very fast classical baseline.

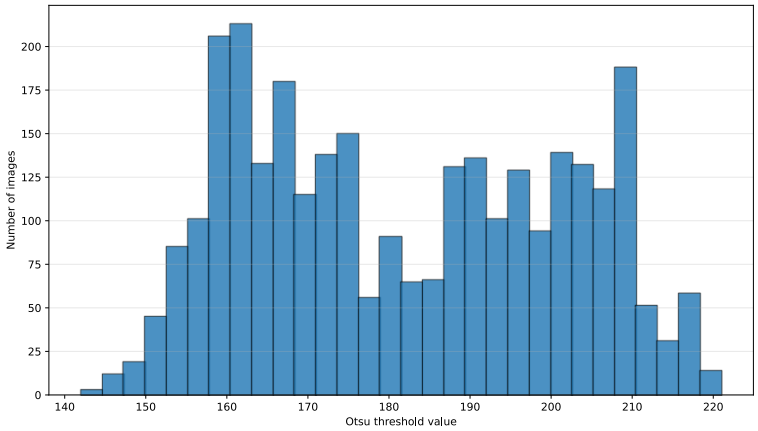

Fig. 9. Distribution of automatically selected Otsu threshold values on the test subset

Fig. 10. Example of Otsu binarization on the test subset with the automatically selected threshold

Watershed Segmentation

For watershed segmentation, the distance threshold ratio for foreground marker generation was selected on the validation subset. The best result was obtained with ratio = 0.4. As shown in **Fig. 11**, increasing the ratio improved Precision but reduced Recall, since stricter foreground markers limited the segmented text regions. The comparison in **Fig. 12** illustrates how different ratio values affect the resulting masks. On the test subset, watershed achieved IoU = 0.567, DSC = 0.718, PA = 0.905, Precision = 0.721, and Recall = 0.756. Compared with K-means and Otsu, watershed produced more balanced Precision and Recall, but it missed more text pixels. The best test example obtained with the selected ratio is shown in **Fig. 13**. The average runtime was 0.111 ms/image, remaining relatively low for a region-based method.

Fuzzy C-means Clustering

For Fuzzy C-means, the number of clusters was fixed to 2, and the fuzziness coefficient was selected on the validation subset. The best result was obtained with m = 1.7. As shown in **Fig. 14**, changing m had only a minor effect on the validation metrics, indicating that the method behaved similarly across the tested fuzziness values. The comparison in **Fig. 15** illustrates how different values affect the resulting masks. On the test subset, Fuzzy C-means achieved IoU = 0.590, DSC = 0.727, PA = 0.885,

Precision = 0.599, and Recall = 0.983. Similar to K-means and Otsu, the method detected almost all text pixels, but also included degraded background regions as foreground. The best test example obtained with the selected parameters is shown in **Fig. 16**. The average runtime was 2.965 ms/image, making it noticeably slower than K-means due to iterative soft membership estimation.

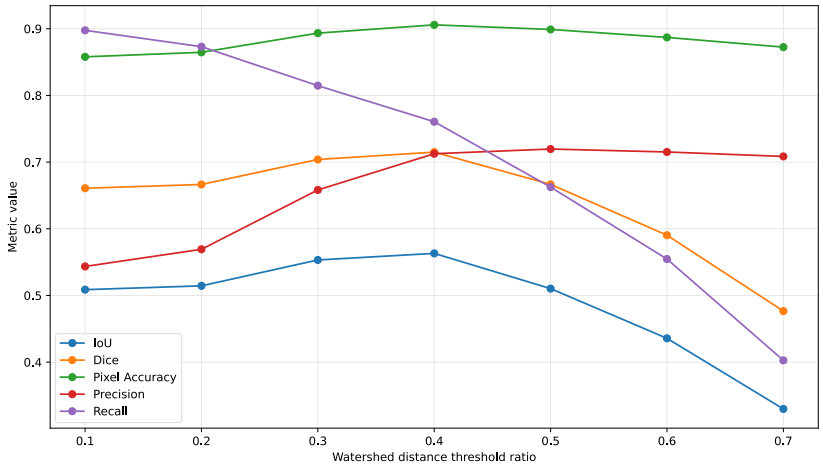

Fig. 11. Effect of the watershed distance threshold ratio on validation metrics

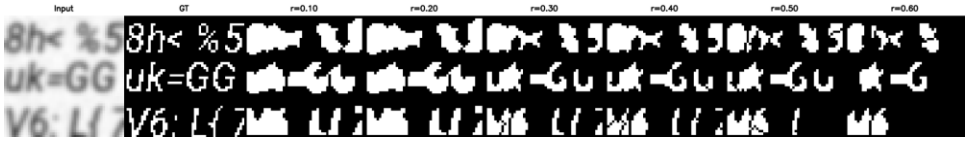

Fig. 12. Visual comparison of watershed results with different distance threshold ratios on a test image

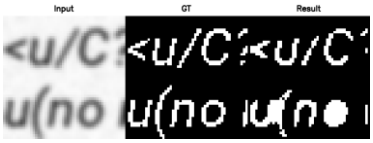

Fig. 13. Example of the best watershed result on the test subset using the validation-selected ratio $r = 0.4$

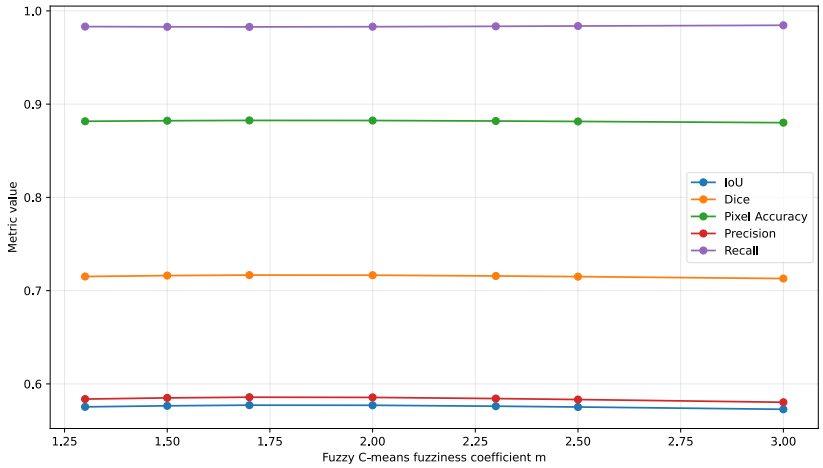

Fig. 14. Effect of the fuzziness coefficient m on validation metrics for Fuzzy C-mean

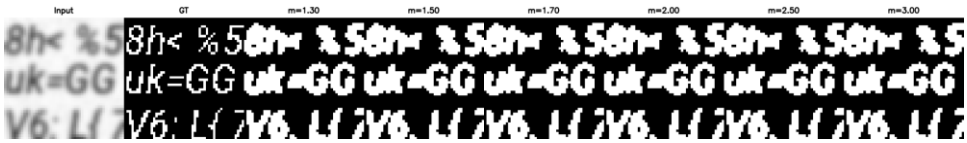

Fig. 15. Visual comparison of Fuzzy C-means results with different *m* values on a test image.

Fig. 16. Example of the best Fuzzy C-means result on the test subset using the validation-selected value *m* = 1.7

Adaptive Thresholding

For adaptive thresholding, the best validation configuration was Gaussian adaptive thresholding with block size = 3 and C=5. As shown in **Fig. 17**, the method was highly sensitive to the local window size and C value, confirming the importance of validation-based parameter selection. The comparison in **Fig. 18** illustrates how different values affect the resulting masks. On the test subset, adaptive thresholding achieved IoU = 0.785, DSC = 0.877, PA = 0.961, Precision = 0.851, and Recall = 0.923. Compared with global thresholding, it provided a better balance between Precision and Recall, indicating stronger robustness to local intensity variation and noise. The best test example obtained with the selected parameters is shown in **Fig. 19**. The average runtime was 0.022 ms/image, which is still very low and suitable for lightweight processing.

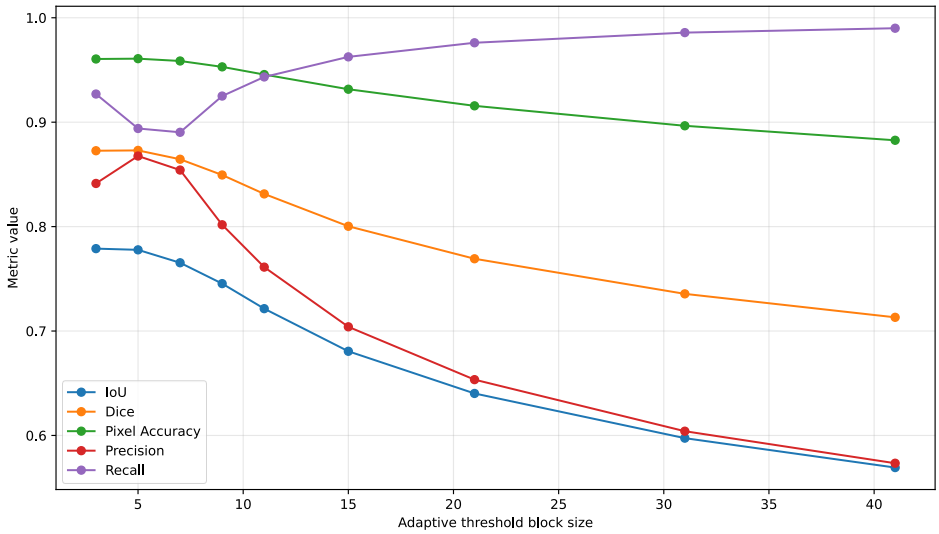

Fig. 17. Effect of adaptive thresholding parameters on validation metrics

Fig. 18. Visual comparison of adaptive thresholding results with different parameter configurations on a test image

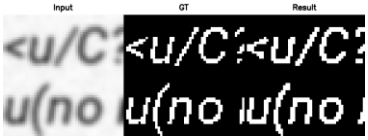

Fig. 19. Example of the best adaptive thresholding result on the test subset using Gaussian adaptive thresholding with block size = 3 and C = 5

Region Growing

For region growing, the growth margin was selected on the validation subset, with the best result obtained at grow_margin = 0. As shown in **Fig. 20**, increasing the growth margin did not improve the segmentation quality, because additional expansion mainly introduced background artifacts rather than useful text pixels. The comparison in **Fig. 21** illustrates how different values affect the resulting masks. On the test subset, the method achieved IoU = 0.590, DSC = 0.727, PA = 0.886, Precision = 0.601, and Recall = 0.978. Similar to clustering-based methods, Region Growing detected almost all text pixels but also included degraded background regions as foreground. The best test example obtained with the selected parameters is shown in **Fig. 22**. The average runtime was 0.073 ms/image, making it relatively fast among region-based methods.

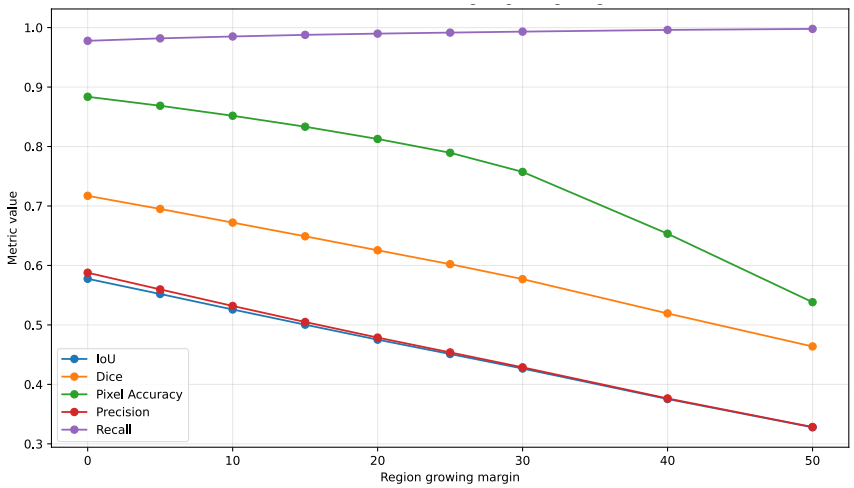

Fig. 20. Effect of the growth margin on validation metrics for Region Growing.

Fig. 21. Visual comparison of Region Growing results with different growth margin values on a test image

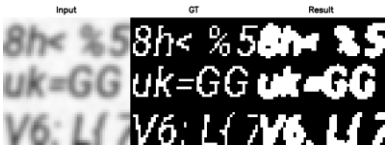

Fig. 22. Example of the best Region Growing result on the test subset using the validation-selected value grow_margin = 0

Region Splitting and Merging

For Region Splitting and Merging, the split threshold was selected on the validation subset, with the best result obtained at split_threshold = 5. As shown in **Fig. 23**, increasing the split threshold reduced IoU, Dice, and Recall because larger regions were no longer divided sufficiently and more text pixels were missed. The comparison in **Fig. 24** illustrates how different values affect the resulting masks. On the test subset, the method achieved IoU = 0.460, DSC = 0.620, PA = 0.830, Precision = 0.500, and Recall = 0.868. The relatively high Recall indicates that much of the text was detected, but the low Precision shows that the region-level decisions also introduced many false foreground pixels. The best test example obtained with the selected parameters is shown in **Fig. 25**. The average runtime was 5.836 ms/image, making it slower than threshold-based and clustering-based methods.

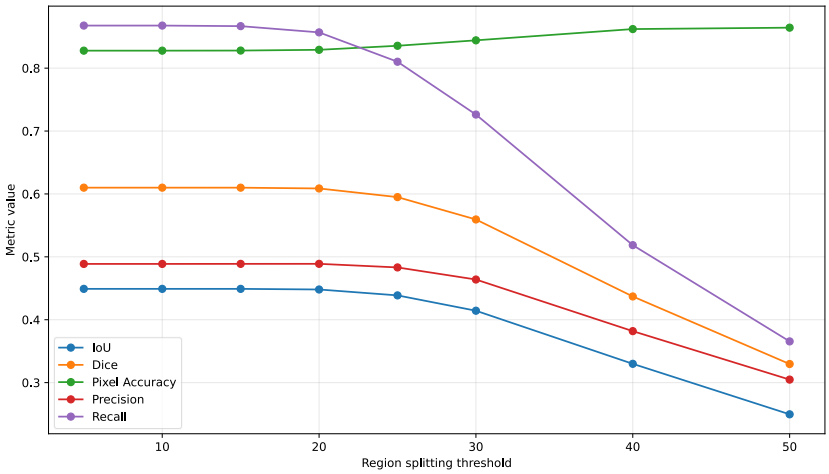

Fig. 23. Effect of the split threshold on validation metrics for Region Splitting and Merging

Fig. 24. Visual comparison of Region Splitting and Merging results with different split threshold values on a test image

Fig. 25. Example of the best Region Splitting and Merging result on the test subset using the validation-selected value split_threshold = 5

Edge-Based Detection

For edge-based detection, the best validation configuration was Canny low threshold = 100, high threshold = 250, and dilation iterations = 1. As shown in **Fig. 26**, changing the Canny low threshold had only a minor effect on the validation metrics. As shown in **Fig. 27**, dilation improved overlap with the ground-truth masks, but the method remained limited because it detects character contours rather than filled text regions. On the test subset, it achieved IoU = 0.313, DSC = 0.469, PA = 0.646, Precision = 0.320, and

Recall = 0.955. The high Recall indicates that most text boundaries were detected, while the low Precision shows that many non-text edges and background artifacts were also included. The best test example obtained with the selected parameters is shown in **Fig. 28**. The average runtime was 0.058 ms/image.

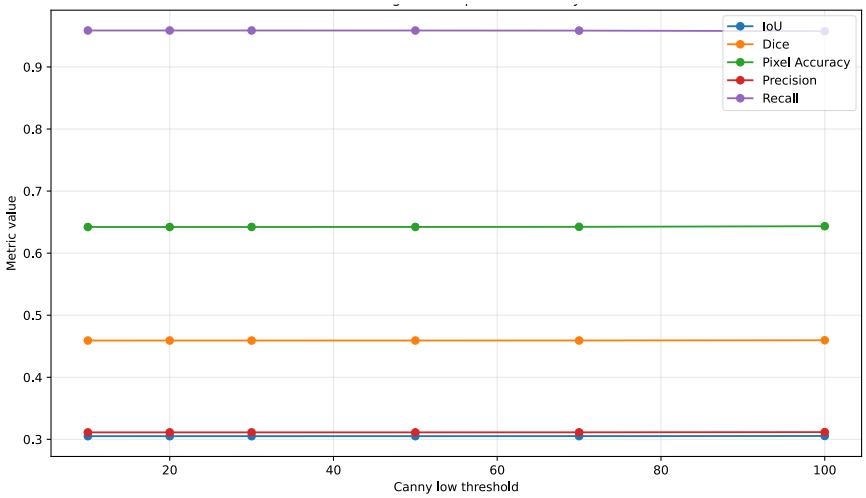

Fig. 26. Effect of Canny edge detection parameters on validation metrics

Fig. 27. Visual comparison of edge-based detection results with different parameter configurations on a test image

Fig. 28. Example of the best edge-based result on the test subset using Canny thresholds 100/250 and one dilation iteration

Laplacian of Gaussian Filtering

For Laplacian of Gaussian filtering, the best validation configuration was Gaussian kernel size = 7, sigma = 1.5, LoG threshold = 40, and dilation iterations = 0. **Fig. 29** shows the influence of LoG threshold on metrics. As shown in **Fig. 30**, the method was sensitive to the response threshold, since low thresholds introduced more noisy edges, while high thresholds removed weak text structures. On the test subset, it achieved IoU = 0.415, DSC = 0.584, PA = 0.823, Precision = 0.492, and Recall = 0.742. These results confirm that LoG captures part of the text structure, but as an edge-based method, it does not fully recover complete foreground regions. The best test example obtained with the selected parameters is shown in **Fig. 31**. The average runtime was 0.036 ms/image.

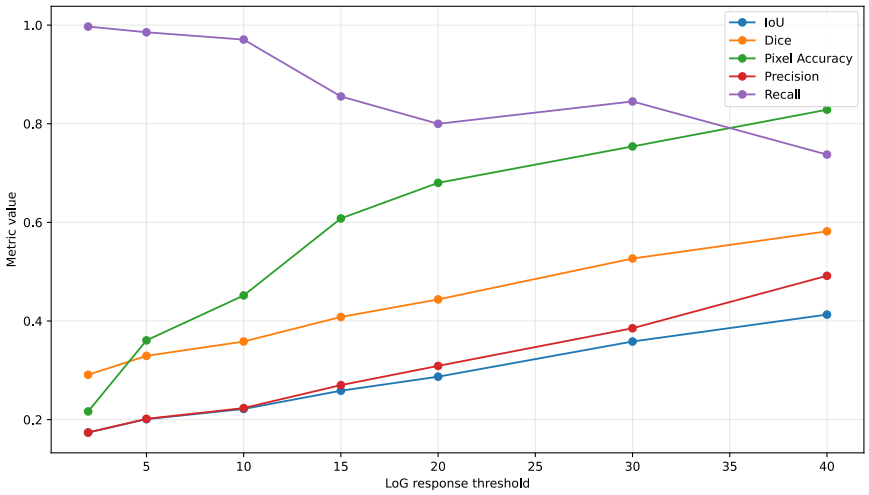

Fig. 29. Effect of Laplacian of Gaussian parameters on validation metrics.

Fig. 30. Visual comparison of Laplacian of Gaussian results with different parameter configurations on a test image

Fig. 31. Example of the best Laplacian of Gaussian result on the test subset using Gaussian kernel size = 7, sigma = 1.5, and LoG threshold = 40

Tiny U-Net

For the neural baseline, the probability threshold was selected on the validation subset, with the best result obtained at threshold = 0.55. The training curve in **Fig. 32** shows that both training and validation loss decreased and stabilized, indicating successful model training. As shown in **Fig. 33**, the validation metrics remained high across a wide threshold range, indicating stable model predictions and good separation between text and background probabilities. The comparison in **Fig. 34** illustrates how different values affect the resulting masks. On the test subset, Tiny U-Net achieved IoU = 0.959, DSC = 0.979, PA = 0.994, Precision = 0.979, and Recall = 0.979. These results substantially outperform the classical methods, showing that the model successfully learned spatial text structure and degraded background patterns. The best test example obtained with the selected threshold is shown in **Fig. 35**. The average inference time was 3.191 ms/image, which is slower than most classical methods but still practical for lightweight processing.

Overall Comparison of Methods

Table 1 summarizes the final test results for all evaluated methods. The reported parameters were selected on the validation subset, while the metrics were computed on the independent test subset.

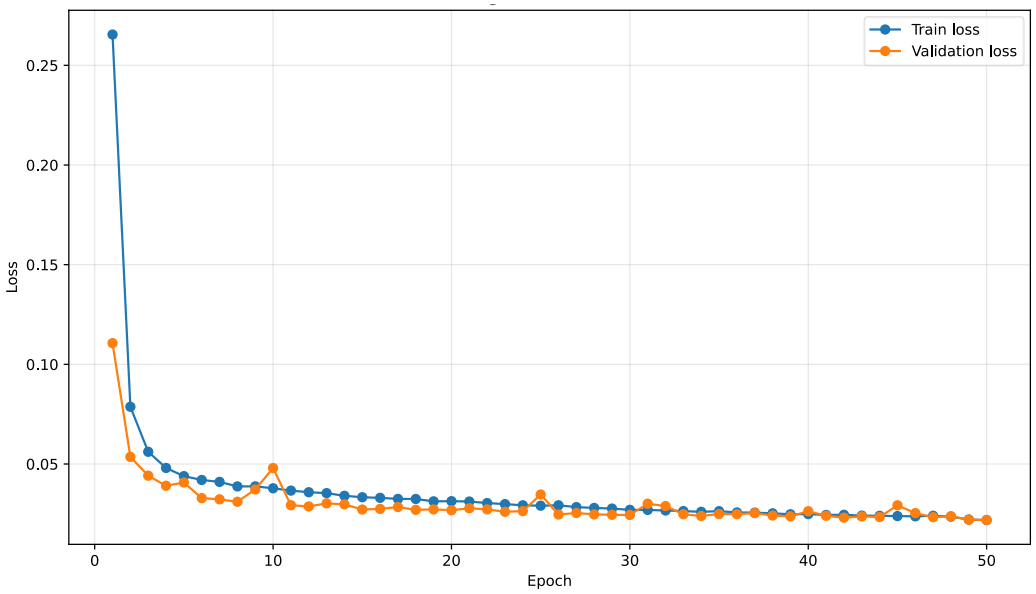

Fig. 32. Training and validation loss during Tiny U-Net training

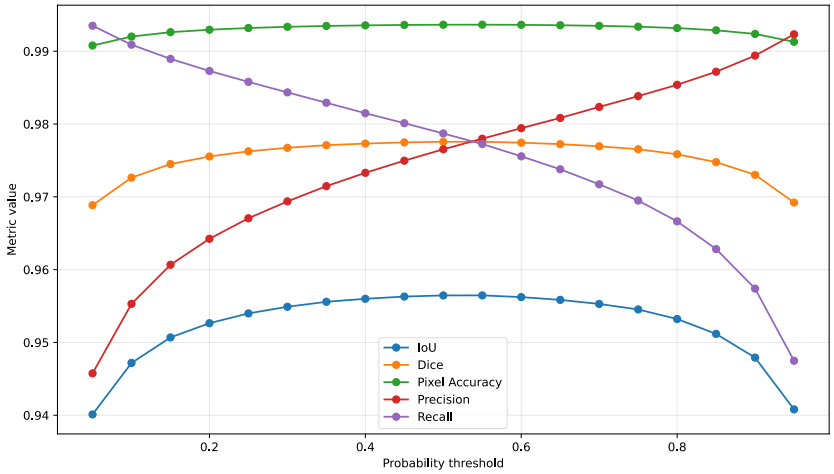

Fig. 33. Effect of the probability threshold on validation metrics for Tiny U-Net

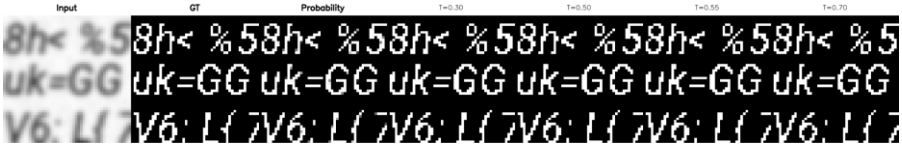

Fig. 34. Visual comparison of Tiny U-Net predictions with different probability thresholds on a test image

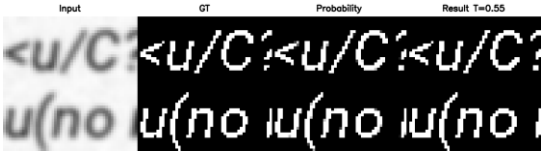

Fig. 35. Example of the best Tiny U-Net result on the test subset using the validation-selected threshold T=0.55

Table 1. **Final comparison of segmentation methods on the test subset.**

Method	Selected parameters	IoU	Dice	Pixel Accuracy	Precision	Recall	Runtime, ms/image
Simple Global Thresholding	(T = 165)	0.661	0.784	0.933	0.751	0.861	0.0015
Otsu Binarization	automatic threshold	0.587	0.725	0.884	0.595	0.984	0.0073
Adaptive Thresholding	Gaussian, block size = 3, (C = 5)	0.785	0.877	0.961	0.851	0.923	0.022
K-means Clustering	(K = 2), max_iter = 5	0.587	0.730	0.886	0.596	0.984	0.305
Fuzzy C-means Clustering	(c = 2), (m = 1.7)	0.590	0.727	0.885	0.599	0.983	2.965
Watershed Segmentation	distance ratio = 0.4	0.567	0.718	0.905	0.721	0.756	0.111
Region Growing	grow_margin = 0	0.590	0.727	0.886	0.601	0.978	0.073
Region Splitting and Merging	split_threshold = 5	0.460	0.620	0.830	0.500	0.868	5.836
Edge-Based Detection	Canny 100/250, dilation = 1	0.313	0.469	0.646	0.320	0.955	0.058
Laplacian of Gaussian	(k = 7), (\sigma = 1.5), threshold = 40, dilation = 0	0.415	0.584	0.823	0.492	0.742	0.036
Tiny U-Net	input size = 50×50, threshold = 0.55	0.959	0.979	0.994	0.979	0.979	3.191

CONCLUSION

This study presented a comparative evaluation of classical segmentation methods and a lightweight neural baseline for binarizing degraded text images. The task was formulated as binary semantic segmentation, where text pixels were treated as the foreground class and background pixels as the background class. To ensure a fair comparison, all tunable parameters were selected on the validation subset, while the final quantitative results were reported on the independent test subset.

The experimental results showed that the evaluated methods differ substantially in terms of segmentation quality, robustness, and computational efficiency. Among the classical approaches, adaptive thresholding achieved the best overall performance, with IoU = 0.785, DSC = 0.877, and PA = 0.961. This confirms that local threshold estimation is more suitable for degraded text images than a single global threshold. At the same time, simple global thresholding demonstrated the lowest runtime, requiring only 0.0015 ms/image, which makes it an attractive option for extremely resource-constrained scenarios despite its lower robustness.

Clustering-based methods, including K-means and Fuzzy C-means, achieved very high Recall values, indicating that they detected most text pixels. However, their lower Precision showed that background degradation and noise were often incorrectly assigned to the foreground class. Region-based methods demonstrated mixed performance. Watershed segmentation provided a more balanced Precision–Recall trade-off, while Region Splitting and Merging suffered from lower foreground accuracy due to coarse region-level decisions. Edge-based methods, including Canny edge detection and Laplacian of Gaussian filtering, were less effective for full-text mask extraction because they primarily detected character contours rather than complete foreground regions.

The Tiny U-Net neural baseline achieved the highest segmentation quality among all evaluated methods, with IoU = 0.959, DSC = 0.979, PA = 0.994, Precision = 0.979, and Recall = 0.979. These results indicate that even a compact neural architecture can effectively learn spatial text structures and distinguish text from degraded background patterns. However, this improvement comes at the cost of higher inference time compared with most classical algorithms.

The results also confirmed that Pixel Accuracy should not be used as the only evaluation metric for text binarization. Due to the dominance of background pixels, Pixel Accuracy can remain high even when the foreground text mask is incomplete or contains many false positives. Therefore, foreground-oriented metrics such as IoU, DSC, Precision, and Recall provide a more reliable interpretation of segmentation quality.

Overall, the results show that different binarization methods are suitable for different application scenarios, depending on the required accuracy, runtime constraints, and available computational resources. Classical methods remain useful when simplicity, interpretability, and low runtime are the main requirements. Adaptive thresholding provides the strongest classical baseline, while Tiny U-Net offers the best segmentation quality when training data and computational resources are available. Future research will focus on extending this comparative evaluation framework to 3D point cloud segmentation and classification, where irregular data structure, noise, varying point density, and complex object geometry introduce additional methodological challenges.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The author received no financial support for the research, writing, and/or publication of this article

COMPLIANCE WITH ETHICAL STANDARDS

The author declares that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

The author has read and agreed to the published version of the manuscript.

REFERENCES

[1] Sauvola, J., & Pietikäinen, M. (2000). Adaptive document image binarization. *Pattern Recognition*, 33(2), 225–236. https://doi.org/10.1016/S0031-3203(99)00055-2

[2] Gatos, B., Pratikakis, I., & Perantonis, S. J. (2006). Adaptive degraded document image binarization. *Pattern Recognition*, 39(3), 317–327. https://doi.org/10.1016/j.patcog.2005.09.010

[3] Sahoo, P. K., Soltani, S., & Wong, A. K. (1988). A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, 41(2), 233-260. https://doi.org/10.1016/0734-189X(88)90022-9

[4] Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. IEEE Transactions on Systems, Man, and Cybernetics, 9(1), 62-66. https://doi.org/10.1109/TSMC.1979.4310076
[5] Bradley, D., & Roth, G. (2007). Adaptive Thresholding Using the Integral Image. *Journal of Graphics Tools*, 12(2), 13-21. https://doi.org/10.1080/2151237X.2007.10129236
[6] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1(14), 281-297. https://projecteuclid.org/euclid.bsmsp/1200512992
[7] Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press. https://doi.org/10.1007/978-1-4757-0450-1
[8] Beucher, S., & Lantuéjoul, C. (1979). Use of Watersheds in Contour Detection. *International Workshop on Image Processing: Real-time Edge and Motion Detection/Estimation*. https://people.cmm.minesparis.psl.eu/users/beucher/publi/watershed.pdf
[9] Adams, R., & Bischof, L. (1994). Seeded Region Growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6), 641-647. https://doi.org/10.1109/34.295913
[10] Horowitz, S. L., & Pavlidis, T. (1974). Picture segmentation by a directed graph of regions and edges. *Proceedings of the 2nd International Joint Conference on Pattern Recognition*, 424-433.
[11] Canny, J. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6), 679-698. https://doi.org/10.1109/TPAMI.1986.4767851
[12] Marr, D., & Hildreth, E. (1980). Theory of Edge Detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 207(1167), 187-217. https://doi.org/10.1098/rspb.1980.0020
[13] Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., & Garcia-Rodriguez, J. (2018). A survey on deep learning techniques for image and video semantic segmentation. *Applied Soft Computing*, 70, 41–65. https://doi.org/10.1016/j.asoc.2018.05.018
[14] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 234–241. https://doi.org/10.1007/978-3-319-24574-4_28
[15] Müller, D., Soto-Rey, I., & Kramer, F. (2022). Towards a guideline for evaluation metrics in medical image segmentation. BMC Research Notes, 15, 210. https://doi.org/10.1186/s13104-022-06096-y
[16] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2), 303-338. https://doi.org/10.1007/s11263-009-0275-4
[17] Dice, L. R. (1945). Measures of the Amount of Ecologic Association Between Species. *Ecology*, 26(3), 297-302. https://doi.org/10.2307/1932409
[18] Fellarrusto. Text Binarization Dataset. Kaggle. Available online: https://www.kaggle.com/datasets/fellarrusto/text-binarization

ЕВОЛЮЦІЯ МЕТОДІВ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ: КЛАСИЧНІ АЛГОРИТМИ ТА ГЛИБИННЕ НАВЧАННЯ

Віктор Вдовиченко
Львівський національний університет імені Івана Франка,
вул. Драгоманова 50, 79005 Львів, Україна

АНОТАЦІЯ

Вступ. Бінаризація текстових зображень є важливим етапом попередньої обробки в задачах аналізу документів, оптичного розпізнавання символів та автоматизованого вилучення інформації. Завдання стає складним, коли зображення містять розмиття, шум, низьку контрастність або деградацію фону.

Матеріали та методи. У цьому дослідженні порівнюються класичні та нейронні методи бінаризації деградованих текстових зображень. Оцінювання проводилося на парному наборі даних для бінаризації тексту, який містить деградовані вхідні зображення та відповідні чисті бінарні цільові маски. Оцінювальні методи включали просте глобальне порогове значення, бінаризацію Оцу, адаптивне порогове значення, кластеризацію K-середніх, метод нечіткої класифікації C-середніх, водороздільна сегментація, нарощування областей, розбиття та злиття областей, детектор границь Кенні, лапласіан гауссіана і модель U-Net. Методи оцінювалися за наступними метриками: коефіцієнтом Жаккара (КЖ), коефіцієнтом подібності Дайса (КПД), точністю пікселів (ТП), влучністю, повнотою та часом виконання.

Результати та обговорення. Адаптивне порогове значення продемонструвало найкращу якість серед класичних методів, досягнувши КЖ = 0,785. Просте глобальне порогове значення було найшвидшим методом із часом виконання 0,0015 мс/зображення, але з нижчою точністю. Метод K-середніх, нечіткий метод C-середніх бінаризація Оцу та нарощування областей продемонстрували високі значення повноти понад 0,97, проте нижчі значення влучності показали, що деградовані ділянки фону часто класифікувалися як передній план. Методи на основі границь показали нижчі значення КЖ та КПД, оскільки вони переважно виявляли контури символів. Модель U-Net досягла найвищої загальної якості: КЖ = 0,959.

Висновки. Результати дослідження показують, що різні методи бінаризації є доцільними для різних умов застосування, залежно від вимог до точності, швидкодії та обчислювальних ресурсів. Класичні методи залишаються привабливими для сценаріїв із низькою затримкою, тоді як легкі нейронні моделі забезпечують суттєво вищу точність сегментації за наявності достатніх обчислювальних ресурсів. Результати також показують, що точність пікселів потрібно інтерпретувати обережно через домінування фонових пікселів у зображеннях документів.

Ключові слова: бінаризація тексту, сегментація зображень, порогове значення, кластеризація, U-Net, показники оцінювання.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
09 April, 2026	25 May, 2026	27 May, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 113–132
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC 004.8; 004.932.4; 623.4

DEEP LEARNING ARCHITECTURES FOR VISUAL RECOGNITION OF SELECTED TYPES OF EXPLOSIVE ORDNANCE: COMPARATIVE ANALYSIS AND SYSTEM IMPLEMENTATION

Volodymyr Hrabovskyi* & ***Oleksii Hryhorashyk***
Department of Optoelectronics and Information Technologies
Ivan Franko National University of Lviv
107 Gen. Tarnavsky Str., Lviv, 79017, Ukraine

Hrabovskyi, V. A., & Hryhorashyk, O. D. (2026). Deep Learning Architectures for Visual Recognition of Selected Types of Explosive Ordnance: Comparative Analysis and System Implementations. *Electronics and Information Technologies*, 34, 113–132. https://doi.org/10.30970/eli.34.7

ABSTRACT

Background. In the context of significant contamination of Ukraine's territory with explosive ordnance (EO) resulting from the Russian Federation's aggression and over 11 years of ongoing hostilities, coupled with limited resources for detection and demining, the implementation of innovative technologies for identifying and recognizing such objects is critical. Therefore, the application of artificial intelligence-based methods for the detection and recognition of explosive devices is both relevant and essential.

Materials and Methods. This article explores the potential of Deep Neural Networks (DNNs) and computer vision for automating EO detection. The study reviews the theoretical foundations and practical applications of Convolutional Neural Networks (CNNs) in object recognition tasks. The primary focus is on utilizing modern deep learning methods to identify explosive devices, specifically technologies employing single-stage (SSD – Single Shot MultiBox Detector, and YOLO – You Only Look Once) and two-stage (R-CNN, Regions-based Convolutional Neural Networks) approaches for object detection and identification. Key aspects of using single- and two-stage DNN architectures for visual object recognition are investigated, comparing the strengths and weaknesses of both approaches and examining ways to improve them to enhance detection efficiency.

Results and Discussion. Based on the conducted analysis, models were developed using single-stage (SSD and YOLOv8) and two-stage (Faster R-CNN) object identification technologies. These models were trained on an original dataset created from internet-sourced images, which included eight common types of EO found in combat zones and de-occupied territories. Testing results of the trained models demonstrated the superiority of the YOLOv8 architecture for explosive device detection and recognition, which was subsequently used to develop a web server for explosive ordnance recognition.

Conclusions. A project was developed to train EO recognition model architectures based on two-stage and single-stage approaches. Using these models, a web server with a REST API (Representational State Transfer Application Programming Interface) was created for EO recognition in both static images and real-time video streams. The proposed system can facilitate demining operations and reduce risks to civilian populations.

Keywords: deep learning, visual recognition, explosive ordnance, SSD (Single Shot MultiBox Detector), YOLOv8, Faster R-CNN

INTRODUCTION

In the context of modern military conflicts, the problem of detecting explosive devices (EDs) has become particularly urgent. For Ukraine, which finds itself at the epicenter of the largest armed confrontation in Europe in recent decades, the timely detection of such objects is critical for both military units and the civilian population. Traditional search methods, which rely on metal detectors and manual operator work, often fail to provide the necessary speed, accuracy, and adaptability required on the modern battlefield.

According to the State Emergency Service of Ukraine, as of late 2024, up to 25% of the country's territory is contaminated with explosive hazards [1]. The frontline and de-occupied areas remain the most dangerous. In recent years alone, Ukrainian sappers have detected and neutralized about 780,000 explosive items [2], yet approximately 144,000 km^2 of territory remains potentially hazardous. Preliminary estimates show that complete clearance will require around 37 billion USD and the involvement of over 10,000 specialists, whereas only about 3,000 sappers currently operate in Ukraine [3]. The scale of the problem is so vast that relying solely on traditional methods could mean that complete clearance takes hundreds of years [4].

One of the most promising avenues for accelerating ED detection is the implementation of computer vision technologies [5], where the vital task is the automatic detection, classification, and localization of objects within images. The primary performance metrics of such systems are recognition accuracy and data processing throughput. According to research [6], the evolution of object detection methods can be broadly divided into two periods: the traditional methods era (before 2014) and the deep learning-based systems era (after 2014) (**Fig. 1**). The rapid advancement of deep learning methods [7] has driven a qualitative breakthrough in the field, transitioning object detection technologies from a primarily research-oriented domain into widespread practical application [8].

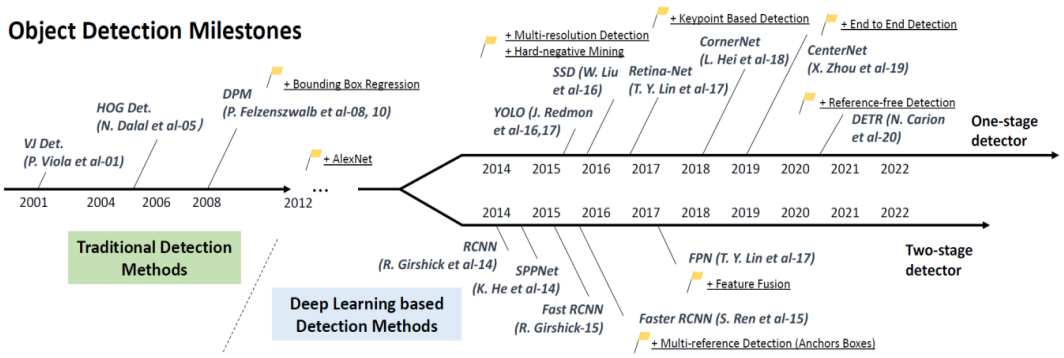

Fig. 1. Object detection "roadmap" [6]

Unlike classical computer vision approaches that require manual feature engineering, deep neural networks can automatically extract informative features directly from raw images. Convolutional neural networks (CNNs) [9, 10] play a pivotal role in this process, demonstrating high efficiency in visual recognition tasks. Their capacity to construct multi-level data representations and identify complex patterns makes these models a promising foundation for automated explosive device detection systems.

The integration of deep learning technologies has led to the formation of two main approaches to object detection: two-stage and one-stage frameworks. Two-stage methods implement a "coarse-to-fine" principle, first generating region proposals and subsequently refining the classification and localization results. One-stage methods execute these operations within a single unified process, offering significantly higher processing speeds and unlocking the potential for real-time applications.

MATERIALS AND METHODS

Two-Stage CNN-Based Recognition Systems

The emergence of two-stage object detection systems was a direct response to the limitations of traditional computer vision methods and early convolutional neural network architectures. Throughout the 2000s and 2010s, CNNs were primarily used for image classification–determining whether an object of a certain class was present without establishing its precise location or quantity. In contrast, the task of object detection requires the simultaneous execution of two interconnected functions: object localization using bounding boxes and its subsequent classification.

A significant challenge was also the handling of objects varying in scale and shape. Traditional approaches often relied on multi-scale analysis or image pyramids, which drastically increased computational costs. The sliding window method [11, 12], where a classifier was repeatedly applied to a vast number of image patches, was particularly resource-intensive. Consequently, these systems could not provide sufficient processing speed for practical use, especially in real-time environments. Additional drawbacks included a heavy reliance on manual feature engineering, insufficient robustness to changing observation conditions, and low efficiency in detecting small or partially occluded objects.

To overcome these constraints, a two-stage approach was proposed, dividing the detection process into two sequential phases. In the first stage, region proposals are generated–potential areas of the image that are likely to contain objects. Fast algorithms, such as Selective Search [13], are utilized here to significantly reduce the number of candidate regions requiring further analysis.

In the second stage, the selected regions are processed by a convolutional neural network, which performs object classification and refines the coordinates of their bounding boxes. By decoupling the localization and recognition tasks, two-stage architectures delivered a substantial leap in detection accuracy and established a foundation for the further evolution of modern computer vision systems.

R-CNN

The development of two-stage object detectors based on convolutional neural networks began with the introduction of the R-CNN (Regions with Convolutional Neural Networks) architecture, proposed by R. Girshick et al. in 2013 [14]. Prior to this, most object detection systems relied on manual feature engineering and the sliding-window method, which required significant computational resources and provided limited accuracy.

The key idea behind R-CNN was to combine region proposal algorithms with the automated feature extraction capabilities of CNNs. The model's workflow consisted of three main stages. First, using the Selective Search algorithm [13], approximately 2,000 candidate regions that could potentially contain objects were generated. Next, each region was scaled to a fixed size, which was required for input into the pre-trained network. In the final stage, the CNN extracted a feature vector, which was then used to determine the presence of an object, classify it, and refine its bounding box coordinates using a regression model.

The introduction of R-CNN delivered a substantial increase in detection accuracy compared to previous approaches and marked a major step forward in the evolution of modern computer vision. At the same time, the architecture suffered from several drawbacks. The most significant of these was the necessity to process each region’s proposal individually. Since a large portion of the candidate regions overlapped, identical features were repeatedly recomputed, which drastically reduced the system's processing speed. Furthermore, the use of fully connected layers required a fixed input size, complicating the handling of regions with different scales and aspect ratios.

To overcome these limitations, the SPPNet (Spatial Pyramid Pooling Network) architecture was proposed [15]. Its primary advantage was that feature maps were computed for the entire image only once and were subsequently shared across all candidate regions. The key element of this model was the Spatial Pyramid Pooling (SPP) layer, which ensured the generation of fixed-size features regardless of the input region's dimensions. This eliminated the need to scale each proposal and significantly reduced processing time compared to R-CNN.

Fast R-CNN and Faster R-CNN

The evolution of two-stage detectors advanced significantly with the introduction of Fast R-CNN [16] (2015) and Faster R-CNN [17] (2015), which brought substantial improvements in both speed and accuracy.

- Fast R-CNN unified the training of all components (feature extraction, classification, and bounding box regression) into a single network. It utilized RoI Pooling (Region of Interest Pooling), a simplified version of SPP, to extract features from object proposals on a feature map generated from the entire image and represent them as a single fixed-length vector.

According to [16], the RoI Pooling mechanism operates by first locating the region on the feature map generated by the backbone CNN that corresponds to each region proposal (RoI). This region, regardless of its arbitrary size, is then divided into a fixed grid (e.g., 7×7 cells). Next, a pooling operation (typically max pooling) is performed on each cell to select the maximum feature value within it. The result of this process is a fixed-size feature vector for each RoI, which can then be fed into fully connected layers for subsequent classification and regression. This approach significantly accelerated both training and inference compared to R-CNN.

- Faster R-CNN marked the next breakthrough by integrating object proposal generation directly into the network using the Region Proposal Network (RPN). The RPN is a fully convolutional network that operates on top of the feature map generated by a backbone convolutional network (such as VGG or ResNet).

One of the key advantages of the RPN is its feature-sharing mechanism, where the features generated by the convolutional layers are simultaneously used by both the RPN and the detection network (Fast R-CNN). This eliminates the need for redundant computations, drastically boosting computational efficiency.

Another critical component is the anchor mechanism, which involves placing a set of predefined bounding boxes (anchor boxes) of various scales and aspect ratios at each spatial position of the feature map. For each anchor, the network performs two tasks: it estimates an objectness score (the probability that an object is present) and performs bounding box regression offsets to refine the position of the bounding box.

Due to its fully convolutional architecture, the RPN can efficiently process images of arbitrary sizes, generating high-quality region proposals based on local features. Furthermore, the capacity for joint training with the Fast R-CNN detector within an end-to-end optimization framework allows for a coordinated improvement in the performance of both system components.

Combined, these characteristics ensure high throughput and efficiency for the RPN, substantially cutting down the time required to generate region proposals and enabling the real-time application of two-stage object detectors. The integration of the RPN into Faster R-CNN resolved the primary bottleneck that hindered the speed and efficiency of previous methods–namely, computationally expensive region proposal generation–and transformed Faster R-CNN into a completely end-to-end system, resulting in unprecedented acceleration and accuracy enhancements.

Feature Pyramid Networks

A significant improvement for two-stage detectors, especially in detecting objects of varying scales, was the introduction of Feature Pyramid Networks (FPN, 2017) [18].

Traditional CNNs extract features from the upper layers, which possess rich semantics but low spatial resolution, making the detection of small objects difficult. FPN addresses this by constructing a multi-level feature pyramid where each layer combines semantically strong features from the upper layers with spatially precise features from the lower layers. This is achieved through:

- Bottom-up pathway: A standard CNN generates a hierarchy of feature maps where, with each subsequent layer, spatial resolution decreases while semantic information becomes more abstract.
- Top-down pathway: Using upsampling operations, features from higher levels are enlarged and merged with the corresponding feature maps from lower levels.
- Lateral connections: These connections transfer information between layers of the same scale, enhancing the overall informativeness of the features.

Thanks to FPN, detectors like Faster R-CNN can efficiently identify objects of any size. An extension of this architecture for instance segmentation tasks is Mask R-CNN [19]. By adding a parallel branch for predicting an object mask alongside the standard bounding box branch, it simultaneously localizes the object and generates its high-quality pixel-level mask.

Single-Stage CNN-Based Recognition Systems

Single-stage detectors based on convolutional neural networks have become a powerful alternative to two-stage systems and are currently one of the key directions in computer vision. This approach to visual object detection is characterized by the ability to perform object detection in a single pass of the network, unlike two-stage architectures that first generate region proposals and then classify them. This approach provides a significant increase in recognition speed, which is critical for real-time applications.

The idea of single-stage detection arose from the need to overcome the computational overhead of two-stage systems. The first breakthrough in this direction was the YOLO (You Only Look Once) model, introduced by Joseph Redmon and colleagues in 2016 [20]. YOLO redefined the object detection task as a regression problem, where a single CNN directly predicts bounding boxes and corresponding class probabilities for all objects in an image by dividing the input image into a grid. This allowed for unprecedented inference speeds, albeit with some loss of accuracy compared to the two-stage models of that time, particularly for small and closely packed objects.

YOLO: Architectures and Applications

YOLO (You Only Look Once) is an object detection approach that fundamentally differs from two-stage methods by formulating detection as a single regression problem [20]. This allows the entire detection pipeline to be performed in a single forward pass of a neural network, ensuring high image processing speed.

The main idea of YOLO is to divide the input image into an S×S grid (e.g., 7×7), where each grid cell is responsible for detecting objects whose centers fall within it. For each cell, the model predicts B bounding boxes (typically 2), each described by coordinates (x, y), dimensions (w, h), and a confidence score indicating both the probability of an object being present and the accuracy of localization. In addition, each cell estimates conditional class probabilities for C classes. As a result, the output tensor has a size of S×S×(B×5+C). For example, with S=7, B=2, and C=20, the output becomes 7×7×30.

The YOLOv1 architecture consists of 24 convolutional layers followed by 2 fully connected layers, enabling effective feature extraction from images. Despite its high speed, early versions had certain limitations: each grid cell could predict only one object, which made it difficult to detect multiple small objects within the same region. Moreover, the use of a single-scale representation and the absence of anchor boxes reduced accuracy for small objects and objects with unusual aspect ratios.

Further development of YOLO significantly improved its performance. YOLOv2 (YOLO9000) [21] introduced anchor boxes, K-means clustering for their selection, batch normalization, and a stronger backbone network, Darknet-19. YOLOv3 [22] added multi-scale predictions and the Darknet-53 architecture with residual connections, improving detection accuracy, especially for small objects.

Subsequent versions focused on balancing speed and accuracy: YOLOv4 [23] integrated modern optimization techniques (CSPDarknet53, PANet, SPP, Mosaic augmentation, CIoU loss); YOLOv5 [24] gained popularity due to its PyTorch implementation and ease of deployment; YOLOv7 [25] introduced E-ELAN and improved scaling strategies; YOLOv8 [26] shifted to an anchor-free approach and supports multi-task learning (detection, segmentation, classification); and YOLOv9 [27] focuses on efficiency improvements through Programmable Gradient Information (PGI) and the GELAN architecture, achieving higher accuracy with reduced computational cost.

SSD and Other Single-Stage Recognition Architectures

Almost simultaneously with YOLO, the SSD (Single Shot MultiBox Detector) architecture was proposed in 2016 [28], further advancing the one-stage detection approach. SSD utilizes multi-scale predictions by adding auxiliary convolutional layers to the backbone network (e.g., VGG-16 or ResNet). This enables object detection at various scales by leveraging feature maps from different levels of the network. The application of default (anchor) boxes with diverse aspect ratios at each feature level substantially enhanced accuracy, particularly for small objects, while maintaining high processing speeds.

Subsequent development of one-stage detectors focused on improving accuracy and training stability without causing a significant drop in performance. RetinaNet [29] (2017) addressed the extreme imbalance between background and foreground (object) examples by introducing Focal Loss, which down-weights the loss assigned to well-classified examples. This allowed the model to achieve accuracy comparable to two-stage systems while maintaining high inference speeds.

EfficientDet [30] (2019) focused on efficiency and scalability, introducing the BiFPN (Bi-directional Feature Pyramid Network) for enhanced multi-scale feature fusion, alongside a compound scaling method to uniformly scale the network's depth, width, and input resolution.

FCOS (Fully Convolutional One-Stage Object Detector) [31], also introduced in 2019, moved away from anchor boxes entirely. Instead, it directly predicts the distance from a given pixel to the boundaries of an object, which simplifies the architecture and increases model flexibility.

In conclusion, despite certain limitations in specific scenarios, one-stage detectors have achieved high accuracy and have become a versatile tool for fast and efficient object detection.

Comparison of Key Characteristics of Two-Stage and Single-Stage Recognition Systems

Modern object recognition systems based on Convolutional Neural Networks (CNNs) have achieved significant progress in solving computer vision tasks. They are generally divided into two main categories: two-stage and one-stage detectors, which differ in architecture, speed, and accuracy.

Two-stage models, such as R-CNN, Fast R-CNN, and Faster R-CNN, first generate region proposals where objects are likely to be located and then perform classification and bounding box refinement. This approach provides high localization accuracy but is computationally more complex and slower.

One-stage detectors, such as YOLO (You Only Look Once), SSD (Single Shot MultiBox Detector), and RetinaNet, perform detection in a single forward pass without a

separate proposal generation stage. They directly predict object coordinates and class labels, enabling significantly higher speed and making them suitable for real-time applications, although sometimes at a slight cost in accuracy compared to the best two-stage models.

At the same time, a convergence of the two paradigms can be observed. Modern one-stage models (YOLOv7, YOLOv8, RetinaNet with Focal Loss) have significantly improved accuracy, while two-stage systems continue to be optimized for speed. Techniques such as Feature Pyramid Networks (FPN), anchor boxes, improved loss functions, and advanced backbone networks are widely used in both approaches, allowing models to be adapted to specific task requirements.

Based on the conducted analysis, YOLOv8, SSD, and Faster R-CNN were selected for studying the recognition of explosive devices from images and video streams. These models were fine-tuned using a specially prepared dataset. The most effective system will be used for further evaluation of automated explosive device detection capabilities.

IMPLEMENTATION

The practical component of the project involves three sequential stages aimed at developing an explosive device recognition system.

- Stage I: Curation of the Labeled Image Dataset. This stage encompasses gathering images of the eight primary types of explosive devices, annotating them to generate baseline ground-truth data, and subsequently exporting and structuring the dataset into formats compatible with the selected model architectures.
- Stage II: Training of the Detection Models. This phase focuses on training the YOLOv8, Faster R-CNN, and SSD models. It involves environment configuration, loading pre-trained weights, and fine-tuning them for the specific task of explosive device detection. The models are then trained on the compiled dataset to optimize detection accuracy, followed by validation and testing on independent data to evaluate precision and inference speed.
- Stage III: Comparative Performance Analysis. The final stage entails a comprehensive evaluation of the model benchmarks to select the most effective architecture for the deployment of the recognition system.

Dataset Development for Training Explosive Object Detection Models

For the effective development of visual object detection systems using deep neural networks (DNNs), the availability of a representative and high-quality annotated training dataset is paramount [32, 33, 34]. Data quality has a critical impact on deep learning outcomes; low-quality data leads to degraded model performance even when using optimal network architectures and advanced training algorithms.

A high-quality dataset provides several vital advantages:

- Accuracy and Strong Generalization Capability: It allows the network to learn true underlying patterns in the data, ensuring high accuracy on new, unseen samples. Conversely, noisy or incorrectly annotated data forces the network to learn false correlations or random noise, leading to overfitting.
- Robustness to Noise and Anomalies: Incorporating diverse conditions (varying lighting, aspect angles, and partial occlusions) makes the model robust against real-world environmental interference.
- Speed and Stability of Convergence: It minimizes the risk of oscillations or a complete failure to converge during training.
- Mitigation of Bias: It prevents systematic errors from reflecting in predictions, which is critical for safety-critical and security applications.
- Optimization of Computational Costs: It reduces the training time required to reach target performance metrics.

However, standardized, publicly accessible image datasets of explosive devices (EDs) are non-existent due to several factors:

- Information Sensitivity: Data regarding EDs is highly critical to national security, meaning most collected repositories remain strictly confidential.
- Complexity and Danger of Data Collection: Documenting EDs in field environments is hazardous, expensive, and requires specialized permits and explosive ordnance disposal (EOD) expertise.
- Object Diversity and Specificity: Significant variations in device type, physical condition, viewing angles, and background clutter demand vast amounts of data to achieve proper generalization.

This forces developers to adopt custom approaches: scraping open-source data (frequently limited by low quality), generating synthetic data via 3D modeling (resource-intensive), or utilizing proprietary corporate datasets. Consequently, custom curation and meticulous annotation of a training dataset is the only viable path to successfully engineering effective DNN-based ED detection systems. This ensures that the training data remains directly relevant to the specific characteristics of the target objects.

Since no ready-made datasets exist for detecting Soviet- and Russian-designed explosive devices, creating a proprietary dataset became necessary. Eight of the most prevalent ED types frequently encountered in combat zones and de-occupied territories of Ukraine were selected:

1. PFM-1 ("Lepestok" / "Petal") – an anti-personnel blast mine;
2. PMN-2 ("Black Widow") – an anti-personnel blast mine;
3. F-1 ("Limonka" / "Lemon") – a defensive hand fragmentation grenade;
4. RGD-5 – an offensive hand fragmentation grenade;
5. RKG-3 – a hand anti-tank grenade;
6. TM-62 – a heavy high-explosive anti-tank blast mine;
7. OZM-72 ("Vidhma" / "Witch" or "Bounding Mine") – a bounding fragmentation anti-personnel mine;
8. MON-50 – a directional fragmentation anti-personnel mine.

Images were collected from open sources using search engines. For each device type, 100 unique samples were downloaded. In total, 800 images were selected for primary training and 200 for hyperparameter experimentation. During collection, care was taken to approximate a normal distribution of image quality: roughly two-thirds consisted of clear photos of objects under various natural conditions to facilitate stable model convergence, while the remaining third comprised low-quality samples with unusual angles, partial occlusions, or sensor noise (an example for the PFM-1 is shown in **Fig. 2**). This distribution ensures the system's resilience to real-world visual interference. All gathered images were standardized to the JPEG format, and file naming conventions were restricted to alphanumeric characters, excluding Cyrillic letters and special symbols.

Annotation and data structuring were carried out using the online service CVAT.io (Computer Vision Annotation Tool) [35] – a widely recognized tool for image and video labeling (**Fig. 3**).

A total of 992 images were manually annotated in CVAT. The workflow included creating a project, defining 8 target class labels, uploading the imagery, tight labeling of each ED with bounding boxes while assigning its respective class, and exporting the annotated data in YOLO and COCO formats.

The data structures of these formats differ substantially in how they encode annotations and bind them to image files:

- COCO Format: Labels for all data splits (train, valid, test) are consolidated into a single *_annotations.coco.json file*. This file contains complete metadata regarding the classes; bounding box coordinates, and references to the corresponding image files.

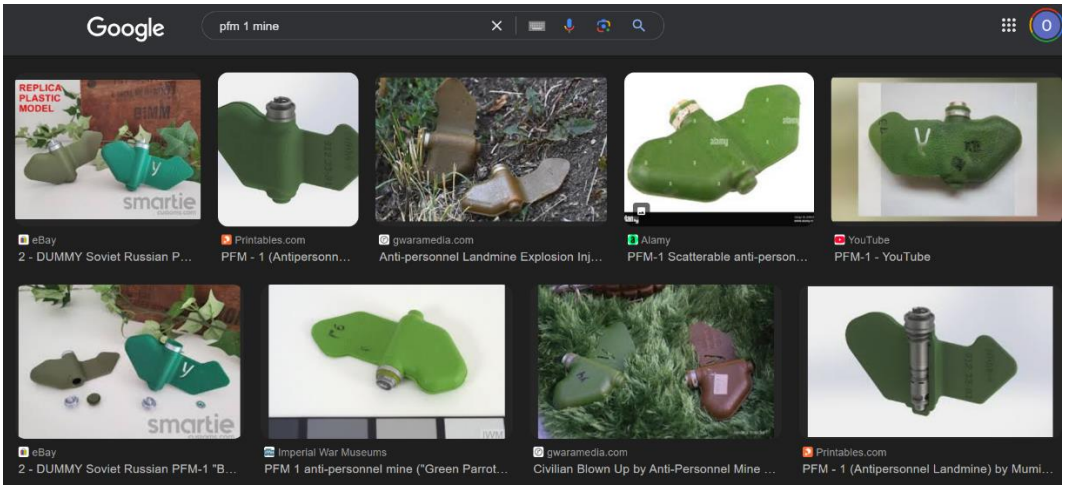

Fig. 2. Visual representation of PFM-1 "Petal" anti-personnel mine images used for dataset creation

Fig. 3. An example of the image annotation process in CVAT

- YOLO Format: Each directory split (train, valid, test) contains two mirrored subdirectories: *images* and *labels*. Each image file maps to a text file with an identical name containing the class indices and normalized bounding box coordinates bounded in the range [0, 1]. Overall metadata, including class names and relative directory paths, is defined in a separate *data.yml* configuration file.

The annotated images were split into two primary datasets:

- explosives_dataset (approximately 800 images) – used for baseline model training.
- explosives_small_dataset (approximately 200 images) – used for rapid hyperparameter optimization, as the smaller data volume allows for faster convergence and accelerated iterative testing.

Each dataset was partitioned into three subsets following an 80% / 10% / 10% ratio:

- Training (train) – 80%: Fed directly into the network to optimize and update the model weights.
- Validation (valid) – 10%: Utilized to monitor performance during training, calculate intermediate evaluation metrics, and guard against overfitting.
- Testing (test) – 10%: Maintained strictly for final benchmarking to evaluate the detector's true performance and generalization capabilities on completely unseen data.

Software Implementation of the Explosive Device Detection System

Structure of the Model Training Project

A software suite was developed for training and evaluating the performance of deep learning models – specifically YOLOv8, SSD+VGG, and Faster R-CNN–to solve the task of explosive device detection in images.

The project architecture includes the following directories:

- explosives_dataset – a directory containing the labeled set of images classified into eight types of explosive devices, divided into training, validation, and test subsets.
- explosives_small_dataset – a reduced version of the main dataset including 200 images; used to verify model convergence and investigate the impact of hyperparameters on the training process.
- trained_models – a directory containing pre-optimized models adapted for the explosive device recognition task, which includes:
  - *exp_yolo.pt* – an optimized model based on YOLOv8;
  - *exp_ssd.pth* – a configuration file for the optimized ssdlite320_mobilenet_v3_large model from the *torchvision* library;
  - *exp_faster_rcnn.pth* – a configuration file for the optimized *fasterrcnn_resnet50_fpn_v2* model from *torchvision*.
- YOLO – a module for training, validation, and practical application of the YOLOv8-based model:
  - yolo_model.ipynb – a Jupyter Notebook for model training and validation;
  - runs – a folder containing training and validation results;
  - *yolo_test.py* – a script for analyzing static images;
  - *webcam_detection.py* – a real-time module for processing video from a webcam.
- R-CNN and SSD – a set of components for training and testing Faster R-CNN and SSD models from the *torchvision* library:
  - RCNN_notebook and SSD_notebook – interactive Jupyter Notebooks for testing the respective models;
  - *pytorch_detection_tools.py* – methods for object detection using these models in PyTorch;
  - *training_tools.py* – tools for training, optimization, and saving models;
  - *faster_rcnn_train.py, ssd_train.py* – scripts for training models on a custom dataset.

Model Training Based on YOLOv8

The script for training and validating the model is contained within the *yolo_model.ipynb* file.

First, we create a model object pre-trained on the COCO dataset:

```
model = YOLO("yolov8n.pt").
```

Alternatively, we load the weights of a saved model that has already been trained on our custom dataset:

```
model_path = "../trained_models/exp_yolo.pt"
model = YOLO(model_path) # using pre-trained model
```

Next, we specify the path to the `data.yaml` configuration file from our dataset:

```
dataset_config = "../explosives_dataset/yolov8/data.yaml"
```

We conduct the model training for 10 epochs, specifying the requirement for input image augmentation:

```
results = model.train(data=dataset_config, epochs=10, augment=True)
```

Finally, we export the model:

```
model.export()
```

After loading the trained model and specifying the path to the validation dataset:

```
model_name = "exp_yolo"
model = YOLO(f"../trained_models/{model_name}.pt")
dataset_config = "../explosives_dataset/yolov8/data.yaml"
```

We perform validation, specifying the relevant metrics:

```
metrics = model.val(data=dataset_config)
metrics.box.map # mAP50-95
metrics.box.map50 # mAP50
metrics.box.map75 # mAP75
metrics.box.maps # a list containing mAP50-95 of each category
```

In addition to the aforementioned mAP50 and mAP50-95, the following metrics are used to evaluate model accuracy: precision (the ratio of correctly identified objects to all identified objects), recall (the ratio of correctly identified objects to all objects that should have been identified), and F1-score (reflecting the balance between precision and recall; used to find the optimal IoU (Intersection over Union) confidence threshold at which the minimum number of Type I and Type II errors is achieved) [37].

The recognition results for all eight types of explosive objects in the validation set are shown in **Table 1.**

Table 1. **Recognition results of the YOLOv8-based model on the validation set objects**

Class	Images	Instances	P	R	mAP50	mAP50-95
all	730	938	0.969	0.956	0.984	0.889
PFM 1	730	191	0.948	0.921	0.978	0.819
PMN 2	730	106	1	0.962	0.994	0.922
F1	730	127	0.956	0.976	0.990	0.867
RGD 5	730	115	0.949	0.974	0.992	0.930
RKG 3	730	44	0.995	0.886	0.947	0.835
TM 62	730	125	0.978	0.968	0.992	0.934
OZM 72	730	127	0.934	0.969	0.987	0.880
MON 50	730	103	0.993	0.990	0.995	0.927

The obtained results demonstrate that the trained YOLOv8 model achieves a mean average precision mAP50 of 0.984. However, the mAP50-95 metric appears more realistic, indicating that in our case, the detected object matches the identified class with an average accuracy of 89%.

A recall (R) value of 0.956 reflects that 95.6% of all objects in the validation dataset were correctly recognized, while a precision (P) of 0.969 indicates that nearly 97% of all recognized objects actually corresponded to the identified class.

Model Training Based on Faster R-CNN

To create the recognition model, the *fasterrcnn_resnet50_fpn_v2* architecture with pre-trained convolutional layer weights was utilized. The modification of the model

involved replacing the existing logistic regressor layers with new ones that provide a number of model outputs corresponding to the number of explosive device types + 1.

A training script, *fast_rcnn_train.py*, was developed, which handles loading the saved model from a configuration file along with optimizer parameters, training epochs, and current error rates, and executes the model training for a specified number of epochs.

To load our dataset, we use a DataLoader for the MS COCO format:

```
train_data_dir = "../explosives_dataset/coco/train"
my_dataset = get_custom_dataset(train_data_dir)
print('Number of samples: ', len(my_dataset))

# DataLoader instance
data_loader = DataLoader(my_dataset,
                         batch_size=2,
                         shuffle=True,
                         collate_fn=collate_fn)
```

We load the *fasterrcnn_resnet50_fpn_v2* model with pre-trained convolutional weights:

```
model = fasterrcnn_resnet50_fpn_v2(pretrained=True)
```

The model modification involved replacing the existing logistic regressor layers at its head with new ones that define the number of model outputs to match the number of detected explosive device types + 1:

```
num_classes = 9  # 8 types of explosive devices + 1 background class
# Update the classifier head
in_features = model.roi_heads.box_predictor.cls_score.in_features
model.roi_heads.box_predictor = torchvision.models.detection.faster_rcnn.
FastRCNNPredictor(in_features, num_classes)
```

The training process can be executed either on a GPU using CUDA or on a CPU:

```
device = torch.device('cuda') if torch.cuda.is_available() else
torch.device('cpu')
model.to(device)
```

SGD (Stochastic Gradient Descent) is used as the optimizer, and StepLR is employed as the learning rate scheduler:

```
params = [p for p in model.parameters() if p.requires_grad]
optimizer = optim.SGD(params, lr=0.001, momentum=0.7, weight_decay=0.004)
lr_scheduler = StepLR(optimizer, step_size=3, gamma=0.1)
```

The model was trained incrementally, with the learning rate decreasing every three scheduler steps:

```
for images, targets in data_loader:
    images = list(image.to(device) for image in images)
    targets = [{k: v.to(device) for k, v in t.items()} for t in targets]

    optimizer.zero_grad()
    losses = model(images, targets)
    loss = sum(loss for loss in losses.values())

    loss.backward()
    optimizer.step()

    if not i % 2:
        print(f"Epoch: {epoch}, Iteration: {i}, Loss: {loss.item()}")
```

```
        i += 1
    lr_scheduler.step()
    epoch += 1
    save_model(model, model_state_path, optimizer, epoch, loss)
```

Additionally, the configuration allows for the use of the Adam optimizer and the ReduceLROnPlateau scheduler:

```
optimizer = optim.Adam(params, lr=0.001)
lr_scheduler = ReduceLROnPlateau(optimizer, mode='min', patience=2,
factor=0.1)
```

SSD Model Training

The model training is implemented in the *ssd_train.py* script. The model is built upon the ssd300_vgg16 architecture, utilizing VGG16 as the backbone convolutional network:

```
def create_ssd_model(num_classes=8, size=640):
    # Load the Torchvision pretrained model.
    model = ssd.ssd300_vgg16()

    # Retrieve the list of input channels.
    in_channels = _utils.retrieve_out_channels(model.backbone, (size, size))

    # List containing number of anchors based on aspect ratios.
    num_anchors = model.anchor_generator.num_anchors_per_location()
```

The modification of this architecture involved replacing the final classification layers as well as the additional SSD convolutional layers directly connected to the logistic regressor:

```
    # The classification head.
    model.head.classification_head = ssd.SSDClassificationHead(
        in_channels=in_channels,
        num_anchors=num_anchors,
        num_classes=num_classes,
    )

    # Image size for transforms.
    model.transform.min_size = (size,)
    model.transform.max_size = size

    return model
```

Adam was used as the optimizer, and ReduceLROnPlateau was employed as the learning rate scheduler:

```
optimizer = optim.Adam(params, lr=0.001)
lr_scheduler = ReduceLROnPlateau(optimizer, mode='min', patience=2,
factor=0.1, verbose=True)
```

Analysis of the Obtained Recognition Results on Validation and Test Sets

Three different types of object detection models were trained on the custom-made explosive device dataset. To evaluate the accuracy of the models, the following metrics were used: mAP50, mAP50-95, precision, and recall [36]. The absolute values of these metrics are a function of the IoU (Intersection over Union) confidence threshold. Therefore, it is necessary to find a threshold value that achieves a satisfactory balance between the precision and recall metrics.

In the case of the YOLOv8-based model, after only 10 training epochs, the values of the corresponding metrics were: precision = 0.950 and recall = 0.951. The F1-curve

demonstrates that the maximum F1-score is reached at a confidence threshold of 0.765 (**Fig. 4**). The obtained values for the accuracy metrics mAP50 and mAP50-95 for the validation and test sets of our dataset are as follows:

- Validation set: mAP50 – 0.984 and mAP50-95 – 0.889.
- Test set: mAP50 – 0.943 and mAP50-95 – 0.795.

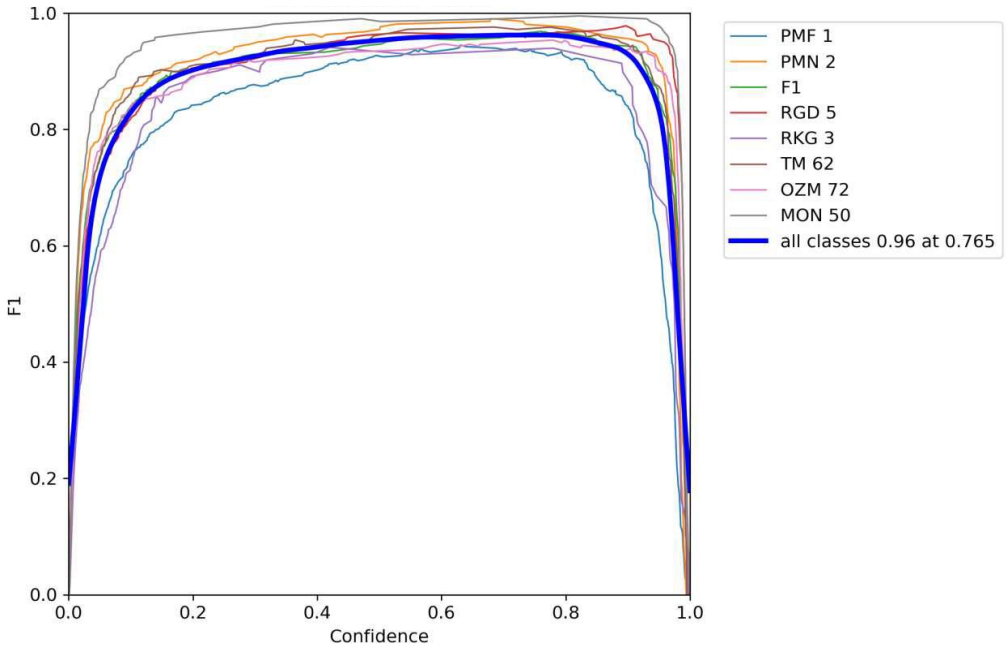

Fig. 4. F1-confidence curve for the YOLOv8 model.

The higher metric values for the validation set in the YOLOv8 model can be explained by the fact that the test set contained more images with interference (noise). Therefore, the test set results are clearly more representative for evaluating the real-world recognition performance of the specified explosive devices by this model.

At the same time, the SSD and Faster R-CNN models showed significantly worse convergence than the YOLOv8-type model. Adjusting the training hyperparameters (switching the optimizer from SGD to Adam and the scheduler from StepLR to ReduceLROnPlateau, which should have provided a more adaptive reduction in the learning rate) did not allow for accuracy comparable to the YOLO model. Specifically, the Faster R-CNN-based model, after a total of approximately 20 training epochs and various manipulations with training parameters and optimizers, was able to reach an mAP50 of 0.724 and an mAP50-95 of 0.421. The SSD-based model encountered similar issues as Faster R-CNN, achieving an mAP50 of 0.630 and an mAP50-95 of 0.261. It was established that under the given conditions, after the third epoch, the SSD and Faster R-CNN models became "stuck" in a local minimum plateau and, regardless of the optimizer parameters or even its type, failed to achieve sufficient accuracy.

Although all three models were trained on the same dataset, it appears that the error function landscape (loss landscape) for them turned out to be different. While the YOLO architecture managed to reach a satisfactory minimum error providing approximately 95% accuracy, the models of the other two architectures failed to achieve a satisfactory result.

These results were somewhat unexpected. According to the data presented in [37], these three models achieve nearly comparable accuracy levels on popular general-purpose datasets, such as COCO and Pascal VOC. However, in our case, when using a

custom-made dataset, the training results for the Faster R-CNN and SSD models failed to achieve satisfactory accuracy that would allow for their integration into an explosive device recognition server. Based on our findings and considering the conclusions of [37], it can be assumed that the training performance of the models depends not only on their architecture but is also significantly influenced by the structure, quality, and specific characteristics of the training dataset, which, in turn, are also determined by the application domain.

Web Server Development for Trained Models

The web server was developed using the Flask framework for Python, a powerful tool for building web applications and APIs. The server's endpoints are Image Detection (*/image-detection*) for recognition in static images and Video Feed (*/video-feed*) for real-time video stream recognition.

Based on the obtained results, the trained YOLOv8 model was selected for the explosive device recognition functionality within the web server. The use of this model for recognition, especially for real-time applications, is recommended by [37].

Experimental deployment of the server was performed on an Apple M4 Max (36 GB RAM) hardware platform with an integrated graphics system lacking CUDA architecture support. Under these conditions, the developed web server delivers an average processing speed of over 30 FPS. Utilizing hardware with CUDA technology support, however, will significantly optimize computational resources and enable system load scaling.

The project consists of:

- three saved model configuration files;
- *detection.py* – containing the detector descriptions and logic;
- *server.py* – defining the endpoints and rendering the user interface and API documentation;
- *swagger.py* – containing the documentation for the endpoints;
- *image.html* and *video.html* – defining the user interface pages, along with *style.css* for the stylesheet.

The "/image-detection" endpoint accepts an image, detects explosive devices within it, overlays bounding boxes and class names, and returns the processed image to the client (**Fig. 5**).

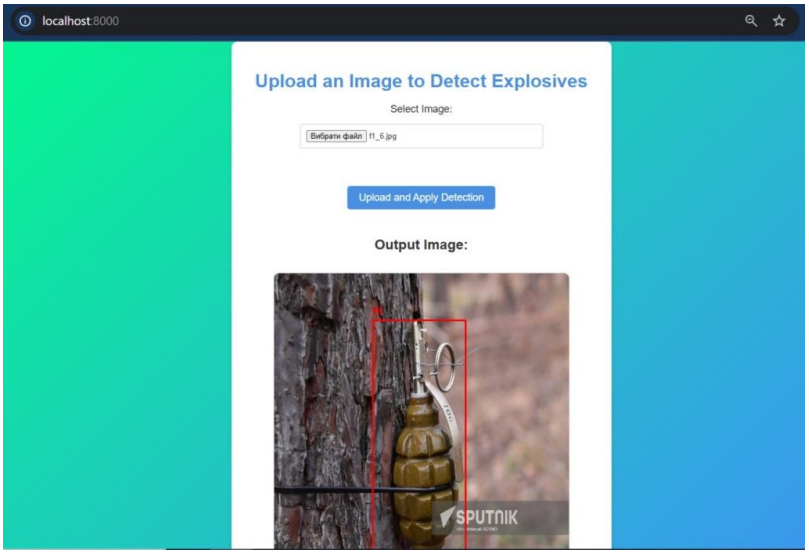

Fig. 5. View of the explosive device recognition window for images.

The "/video-feed" endpoint receives a video stream, detects explosive devices in its frames, overlays bounding boxes and class names, and returns the processed video frames to the client in real-time (**Fig. 6**).

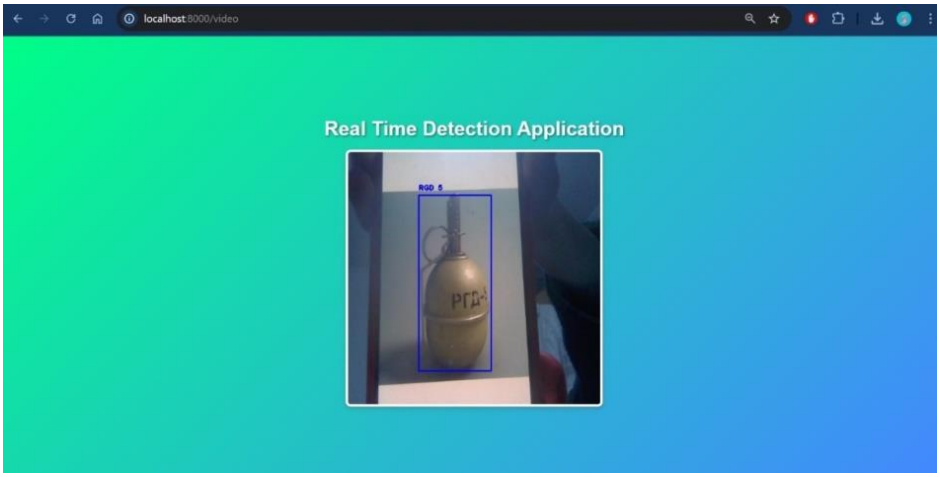

Fig. 6. Real-time recognition result from a video stream.

CONCLUSIONS

This study confirms the effectiveness of applying deep neural networks (DNNs) to the practical task of automated explosive device detection in both static images and video streams. The developed web server with a REST API ensures the integration of trained deep learning models into application systems, enabling the detection of hazardous objects in uploaded images as well as in real-time mode.

In the course of this work, a theoretical analysis of modern image processing methods and deep neural network architectures was conducted, specifically focusing on single-stage and two-stage approaches (YOLOv8, SSD, Faster R-CNN). To train the models, a proprietary dataset was formed containing images of the eight most common types of explosive devices found in Ukraine. The created dataset facilitated proper training and enabled a comparative analysis of the selected architectures.

Experimental results demonstrated that:

- The YOLOv8 model exhibited the highest performance metrics (mAP50 = 0.984 on validation data and 0.943 on test data; maximum F1-score reached at a threshold of 0.765), indicating its suitability for the stated task.
- The Faster R-CNN and SSD models proved to be less effective when trained on our custom dataset (mAP50-95 = 0.421 and 0.261, respectively). This could be attributed to architectural limitations, specific characteristics of the created dataset, and the insufficient robustness of these models to data variability.

Based on the most productive model under our conditions (YOLOv8), a software suite was implemented capable of providing efficient and rapid explosive device detection across various use cases. The interface, built in accordance with the OpenAPI standard, offers users flexible options for system integration and practical application.

The results obtained confirm the high potential of deep neural networks for enhancing security levels and demonstrate the feasibility of developing robust practical solutions based on modern deep learning methods. Future research should focus on expanding the dataset, optimizing models for operation on resource-constrained devices, and integrating the system with other security monitoring technologies.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [V.H.]; methodology, [H.V, O.H.]; validation, [O.H.]; writing – original draft preparation, [V.H.]; writing – review and editing [H.V, O.H.]; supervision, [V.H.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Interactive map of areas potentially contaminated with explosives. (ukr) URL: https://mine.dsns.gov.ua/

[2] What percentage of Ukraine's territories are mined? (ukr). URL: https://tsn.ua/ukrayina/skilki-vidsotkiv-teritoriy-ukrayini-zaminovani-klimenko-prigolomshiv-cifroyu-2550241.html

[3] Over 144,000 sq km of Ukraine are considered potentially mined – new data from the Ministry of Internal Affairs. (ukr). URL: https://suspilne.media/781039-ponad-144-tisaci-kv-km-ukraini-vvazautsa-potencijno-zaminovanimi-novi-dani-mvs/

[4] Demining Ua – Rozminuvannya Ukraiyy (ukr). URL: https://demininguа.com/rozminuvannya-ukrayini

[5] Szeliski, R. (2022). *Computer vision: Algorithms and applications* (2nd ed.). Springer. https://doi.org/10.1007/978-3-030-34372-9

[6] Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. *Proceedings of the IEEE,* 111(3), 257–276. https://doi.org/10.48550/arxiv.1905.05055

[7] Lamichhane, B. R., Srijuntongsiri, G., & Horanont, T. (2025). CNN based 2D object detection techniques: A review. *Frontiers in Computer Science*, 7, 1437664. https://doi.org/10.3389/fcomp.2025.1437664

[8] Edozie, E., et al. (2025). Comprehensive review of recent developments in visual object detection based on deep learning. *Artificial Intelligence Review*, 58, 277. https://doi.org/10.1007/s10462-025-11284-w

[9] Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018, Article 7068349. https://doi.org/10.1155/2018/7068349

[10] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM,* 60(6), 84–90. https://doi.org/10.1145/3065386

[11] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2014). OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *In International Conference on Learning Representations (ICLR 2014).* https://arxiv.org/abs/1312.6229

[12] Szegedy, C., Toshev, A., & Erhan, D. (2013). *Deep neural networks for object detection*. In C. J. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems* (Vol. 26). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2013/file/f7cade80b7cc92b991cf4d2806d6bd78-Paper.pdf.

[13] Uijlings, J. R. R., van de Sande, K. E. A., Gevers, T., & Smeulders, A. W. M. (2013). Selective search for object recognition. *International Journal of Computer Vision*, 104(2), 154–171. https://doi.org/10.1007/s11263-013-0620-5
[14] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587). https://doi.org/10.1109/CVPR.2014.81
[15] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, *37*(9), 1904-1916. https://doi.org/10.1109/TPAMI.2015.2389824
[16] Girshick, R. (2015). Fast R-CNN. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440-1448). https://doi.org/10.1109/ICCV.2015.169
[17] Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, *39*(6), 1137-1149. https://doi.org/10.1109/TPAMI.2016.2577031
[18] Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2117–2125). https://doi.org/10.1109/CVPR.2017.106
[19] He, K.M., Gkioxari, G., Dollár, P. and Girshick, R. (2017) Mask R-CNN. *IEEE International Conference on Computer Vision (ICCV)*, Venice, 22-29 October 2017, 386-397. https://doi.org/10.1109/ICCV.2017.322
[20] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 779–788). IEEE. https://doi.org/10.1109/CVPR.2016.91
[21] Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 7263–7271). IEEE. https://doi.org/10.1109/CVPR.2017.690
[22] Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint* arXiv:1804.02767. https://doi.org/10.48550/arXiv.1804.02767
[23] Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint* arXiv:2004.10934. https://arxiv.org/abs/2004.10934
[24] Jocher, G. (2020). YOLOv5 by Ultralytics *(Version 7.0*) [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.3908559
[25] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies for real-time object detection. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 7701–7711). https://doi.org/10.48550/arXiv.2207.02696
[26] Ultralytics. (2023). YOLOv8 Official Documentation. Retrieved from https://docs.ultralytics.com/yolov8/
[27] Wang, C. Y., Liao, H. Y. M., & Yeh, I. H. (2024). YOLOv9: Learning what you want to learn using programmable gradient information. *arXiv preprint* arXiv:2402.13616. https://arxiv.org/abs/2402.13616
[28] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer vision – ECCV 2016*. Lecture Notes in Computer Science (Vol. 9905, pp. 21–37). Springer, Cham. https://doi.org/10.1007/978-3-319-46448-0_2

[29] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 2977–2985). IEEE. https://doi.org/10.48550/arXiv.1708.02002
[30] Tan, M., Pang, R., & Le, Q. V. (2020). EfficientDet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 10781–10790). IEEE. https://doi.org/10.48550/arXiv.1911.09070
[31] Tian, Z., Shen, C., Chen, H., & He, T. (2019). FCOS: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 9627–9636). IEEE. https://doi.org/10.48550/arXiv.1904.01355
[32] Domingos, P. (2015). *The master algorithm: How the quest for the ultimate learning machine will remake our world.* Basic Books.
[33] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
[34] Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer.
[35] CVAT.io – Open Data Annotation Platform. URL: https://www.cvat.ai/
[36] mAP (mean Average Precision) for Object Detection, URL: https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173
[37] Shobaki, W. A., & Milanova, M. (2025). A comparative study of YOLO, SSD, Faster R-CNN, and more for optimized eye-gaze writing. *Sci*, 7(2), 47. https://doi.org/10.3390/sci7020047

АРХІТЕКТУРИ ГЛИБОКОГО НАВЧАННЯ ДЛЯ ВІЗУАЛЬНОГО РОЗПІЗНАВАННЯ ВИБРАНИХ ТИПІВ ВИБУХОНЕБЕЗПЕЧНИХ ПРЕДМЕТІВ: ПОРІВНЯЛЬНИЙ АНАЛІЗ ТА СИСТЕМНІ ВПРОВАДЖЕННЯ.

Володимир Грабовський*, Олексій Григорашик
Кафедра оптоелектроніки та інформаційних технологій,
Львівський національний університет імені Івана Франка,
вул. Ген. Тарнавського 107, Львів, 79017, Україна

АНОТАЦІЯ

Вступ. У контексті значного забруднення території України вибухонебезпечними пристроями (ВНП), що сталося внаслідок агресії Російської Федерації та понад 11 років бойових дій, а також обмежених ресурсів для виявлення та розмінування, впровадження інноваційних технологій для ідентифікації та розпізнавання таких об'єктів є критично важливим. Тому застосування технологій на основі штучного інтелекту для виявлення та розпізнавання вибухових пристроїв є актуальним та необхідним.

Матеріали та методи. У цій статті досліджується потенціал глибоких нейронних мереж (ГНМ) та комп'ютерного зору для автоматизації виявлення ВНП. У дослідженні розглядаються теоретичні основи та практичне застосування згорткових нейронних мереж (ЗНМ) у задачах розпізнавання об'єктів. Основна увага приділяється використанню сучасних методів глибокого навчання для ідентифікації вибухових пристроїв, зокрема технологій, що використовують одноступінчасті (однопрохідний детектор об'єктів із множинними обмежувальними рамками SSD – Single Shot MultiBox Detector, та однопрохідний метод виявлення об'єктів YOLO – You Only Look Once,) та двоступінчасті (регіонально-орієнтована згорткова нейронна мережа R-RCN) підходи для виявлення та ідентифікації об'єктів. Досліджено ключові аспекти використання одно- та двоступінчастих архітектур ГНМ для візуального

розпізнавання об'єктів, порівняння сильних та слабких сторін обох підходів, а також вивчення шляхів їх удосконалення для підвищення ефективності виявлення.

Результати. На основі проведеного аналізу було розроблено моделі з використанням одноступінчастої (однопрохідний детектор із множинними обмежувальними рамками SSD, та одноступінчастий детектор виявлення об’єктів, восьма версія архітектури You Only Look Once YOLOv8) та двоступінчастої (двоступінчастий регіональний детектор об’єктів на основі ЗНМ Faster R-CNN) технологій ідентифікації об'єктів. Ці моделі були навчені на оригінальному наборі даних, створеному з зображень з Інтернету, який включав вісім поширених типів ВНП, що зустрічаються в зонах бойових дій та на деокупованих територіях. Результати тестування навчених моделей продемонстрували перевагу архітектури YOLOv8 для виявлення та розпізнавання вибухових пристроїв, що було використано для розробки веб-сервера для розпізнавання вибухонебезпечних пристроїв.

Висновки. Розроблено проєкт для навчання моделей розпізнавання ВНП на основі двоступінчастого та одноступінчастого підходів. Використовуючи ці моделі, створено веб-сервер з REST API (інтерфейсу прикладного програмування на основі архітектури Representational State Transfer) для розпізнавання ВНП як на статичних зображеннях, так і у відеопотоках у реальному часі. Запропонована система може сприяти прискоренню процесів розмінування та зменшенню ризиків для цивільного населення.

Ключові слова: глибоке навчання, візуальне розпізнавання, вибухонебезпечні предмети, SSD (Single Shot MultiBox Detector), YOLOv8, Faster R-CNN

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
28 May, 2026	02 Juny, 2026	08 Juny, 2026	29 Juny, 2026

Electronics and Information Technologies, 2026, 34, 133–144
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.85

ENSEMBLE MACHINE LEARNING TECHNIQUES FOR GENOMIC VARIANT PATHOGENICITY PREDICTION

Andrii Smoliak* , ***Ivan Kuno***
Department of Optoelectronics and Information Technologies,
Ivan Franko National University of Lviv,
107 Gen. Tarnavskoho Str., Lviv, 79017, Ukraine

Smoliak, A., Kuno, I. (2026). Ensemble Machine Learning Techniques for Genomic Variant Pathogenicity Prediction. *Electronics and Information Technologies*, 34, 133–144. https://doi.org/10.30970/eli.34.8

ABSTRACT

Background. The rapid development of next-generation sequencing (NGS) technologies has resulted in a massive accumulation of genomic data, creating new opportunities and challenges for identifying genetic variants associated with diseases. Traditional statistical and rule-based approaches often fail to capture the complex, non-linear relationships between genomic features and variant pathogenicity. The integration of ensemble machine learning techniques provides a promising pathway to enhance prediction accuracy and interpretability in genomic research. This study focuses on developing an ensemble-based framework for the classification of genetic variants using gradient boosting.

Materials and Methods. The dataset was obtained from the open-access ClinVar repository, containing annotated genomic variants with clinical significance labels. After preprocessing and feature engineering, a set of ensemble algorithms, including HistGradientBoosting, XGBoost, and LightGBM, was trained and evaluated. The implemented pipeline was designed to be reproducible and computationally efficient, enabling model training on standard workstation environments.

Results and Discussion. The conducted experiments demonstrated the effectiveness of the proposed framework for the classification of pathogenic and benign genetic variants. The results indicate that the integration of biologically informed genomic features contributes to improved predictive accuracy, robustness, and generalization across heterogeneous genomic datasets. The combination of an expanded feature space with gradient boosting ensemble algorithms achieved strong ROC-AUC and PR-AUC performance on an independent test.

Conclusion. Ensemble learning represents an effective and interpretable approach for genomic variant classification and pathogenicity prediction. The study demonstrated that the integration of biologically informed feature engineering with gradient boosting ensemble models improves predictive robustness, generalization, and resilience to class imbalance in heterogeneous genomic datasets. Future work will focus on advanced deep learning components and further optimization strategies to enhance predictive performance on large-scale genomic datasets.

Keywords: ensemble learning, genomic prediction, pathogenic variants, gradient boosting, feature engineering, bioinformatics.

INTRODUCTION

The rapidly decreasing cost and increasing throughput of next-generation sequencing (NGS) technologies have revolutionized modern genomics. Millions of novel variants are identified annually, yet determining their clinical significance remains a major challenge [1]. Manual curation and laboratory validation are often infeasible at scale, prompting the integration of computational approaches – particularly machine learning (ML) – to automate variant interpretation [2].

ML techniques have demonstrated strong performance across numerous genomics tasks, including variant pathogenicity assessment [3]. Among ML algorithms, ensemble learning and particularly gradient-boosted decision trees (GBDT) have gained attention due to their high predictive performance, robustness to noise, and interpretability [4]. Gradient boosting approaches such as XGBoost have been successfully adopted for genomic variant classification tasks. For instance, a study by Jain et al. demonstrated high accuracy of XGBoost on large variant datasets [5]. A significant motivation for computational pathogenicity prediction is the growing proportion of variants classified as “Variants of Uncertain Significance” (VUS). A substantial proportion of ClinVar entries fall into this category, underscoring the lack of functional evidence and the urgent need for robust computational methods capable of assisting in variant prioritization. Accurate and interpretable prediction models provide meaningful support for clinicians and geneticists, enabling more efficient triage of candidate variants for experimental validation [6].

Previous studies have applied ML to variant impact prediction. CADD introduced a large-scale integrative approach [3], while deep learning models like DANN [7] and consensus methods such as REVEL improved accuracy by aggregating outputs from multiple models. Recent ML-based frameworks further improved automated variant prioritization and classification performance [5]. Nevertheless, many existing frameworks are highly gene-specific, dependent on extensive protein-level annotations, or require high-performance computing resources [8]. There remains a need for a lightweight, interpretable, and reproducible machine learning pipeline capable of high-accuracy pathogenicity prediction using open-access genomic datasets. The present study proposes such a framework, combining gradient boosting models with extended feature engineering to improve discriminatory performance and biological interpretability. The ultimate purpose of this study is to design and implement an interpretable ensemble-based model that integrates feature engineering and boosting algorithms for effective genomic variant classification. The primary contribution of this work is not the introduction of a novel machine learning algorithm, but the design and evaluation of a reproducible computational pipeline for genomic variant pathogenicity prediction that integrates biologically informed feature engineering, automated Bayesian hyperparameter optimization, and gradient boosting ensemble algorithms (HistGradientBoosting, XGBoost, and LightGBM). Particular emphasis is placed on the systematic construction of an expanded biologically informed feature space and on evaluating its contribution to predictive robustness, generalization, and pathogenicity prediction performance.

MATERIALS AND METHODS

The Methodology section provides a detailed description of the lightweight and reproducible machine learning pipeline developed for the variant classification task, ensuring that all procedures can be reliably replicated by other researchers.

Data Acquisition and Preprocessing

The foundational dataset for model training was the ClinVar repository, an open-access resource containing annotated genomic variants paired with their clinical significance labels. The analysis focused on binary classification: distinguishing between pathogenic/likely pathogenic variants and benign/likely benign variants. Variants labeled as 'Uncertain Significance' (VUS) were excluded to maintain annotation purity. The final dataset consisted of 58,214 clinically annotated variants extracted from ClinVar, including 12,487 pathogenic or likely pathogenic variants and 45,727 benign or likely benign variants. After preprocessing and removing entries with incomplete annotations, 56,902 variants remained for model development. This distribution reflects the characteristic imbalance of real-world genomic datasets, underscoring the need for metrics and algorithms tailored for rare-class prediction.

Data Splitting and Experimental Design. The dataset was subjected to comprehensive preprocessing, including filtering variants based on clear clinical significance and handling missing feature values. Missing continuous values (e.g., allele frequency) were imputed using feature-wise means, while categorical variables (e.g., variant impact) were imputed via mode. The processed dataset was then randomly partitioned into three subsets: a training set (70%), a validation set (15%), and a testing set (15%), ensuring independent evaluation of the model's generalization capacity.

Pipeline Architecture and Workflow

The proposed computational workflow was implemented as a domain-specific, modular, and reproducible pipeline for genomic variant pathogenicity prediction, as illustrated in **Fig. 1**. Unlike a generic data-processing workflow, the pipeline reflects the main stages required for transforming raw ClinVar annotations into a machine-readable feature matrix and evaluating pathogenicity prediction models.

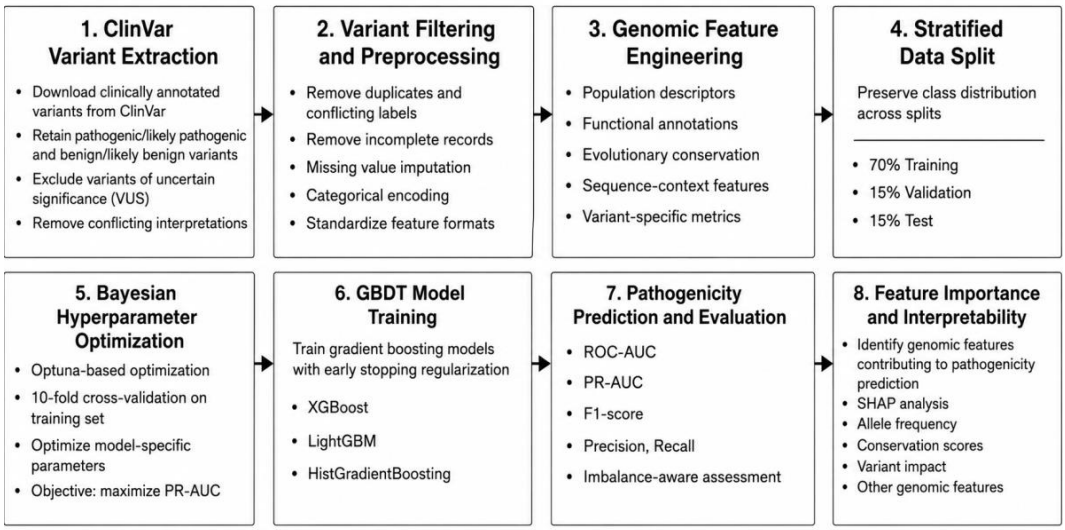

Fig. 1. Domain-specific computational pipeline for genomic variant pathogenicity prediction. The pipeline illustrates the transformation of clinically annotated ClinVar variants into an expanded biologically informed genomic feature space for pathogenicity prediction

At the data acquisition stage, clinically annotated variants were extracted from the ClinVar repository. Variants with uncertain or conflicting clinical significance were

excluded, while pathogenic/likely pathogenic and benign/likely benign variants were retained for binary classification. During preprocessing, incomplete records were removed, missing allele-frequency values were imputed, and categorical annotations such as variant consequence and clinical significance labels were encoded into numerical representations.

At the feature engineering stage, biologically informed descriptors were extracted and integrated into an expanded genomic feature space. These descriptors included population-level features (allele frequency), functional annotations (variant impact categories), evolutionary conservation scores, sequence-context characteristics, and transition/transversion metrics. This stage represented a key component of the proposed framework, enabling the incorporation of biologically relevant information into the predictive models and contributing to improved predictive robustness and generalization across heterogeneous genomic datasets.

The model training stage included Bayesian hyperparameter optimization, early stopping, and ensemble learning with HistGradientBoosting, XGBoost, and LightGBM classifiers. Finally, model performance was evaluated using ROC-AUC and PR-AUC metrics on an independent test set.

The entire workflow was implemented using open-source Python libraries and designed as a lightweight, reproducible framework that can be efficiently executed on standard workstation environments without specialized GPU infrastructure.

Feature engineering was performed to extract biologically relevant predictive signals from the raw variant data. The minimal set of key predictive features included: allele frequency (leveraging population data from resources like 1000 Genomes) [1], variant impact (e.g., missense, loss-of-function), and transition/transversion ratios.

Building upon recent advancements in feature fusion, the study utilized an extended feature space to integrate heterogeneous genomic information. This integrated information incorporated features derived primarily from genomic annotations, including conservation scores, sequence-context descriptors, and functional variant annotations [3], significantly enhancing the model's ability to capture complex non-linear relationships.

The key genomic features used for model training and their biological interpretation are summarized in **Table 1**.

Table 1. **Key Genomic Features Used for Pathogenicity Classification and Their Sources**

Feature Category	Feature Name	Description	Data Type	Example Annotation Tool
Population Genetics	Allele Frequency	Frequency of the variant allele in global or reference populations.	Continuous	gnomAD, 1000 Genomes
Sequence Context	Transition/ Transversion Ratio	Measure of substitution types (Purine/Pyrimidine changes).	Continuous	
Functional Impact	Variant Impact	Predicted consequence on the resulting protein (e.g., Missense, Loss-of-Function).	Categorical	SIFT, PolyPhen-2
Structural Features	Evolutionary Conservation Score	Degree of sequence preservation across multiple species.	Continuous	CADD Score

Ensemble Model Training and Statistical Analysis

The core classification framework relies on advanced gradient boosting decision trees (GBDT) due to their superior performance and interpretability [4]. We trained and evaluated three specific ensemble algorithms: HistGradientBoosting, XGBoost, and LightGBM [5]. This GBDT approach naturally mitigates the challenge of class imbalance by aggregating diverse weak learners, reducing the need for explicit resampling techniques such as SMOTE [6].

Hyperparameter Optimization and Reproducibility. To achieve optimal generalization and efficiency, the training process utilized automated optimization techniques. Bayesian search was employed for hyperparameter tuning [9], and early stopping criteria were implemented to monitor performance on the validation set and prevent overfitting. The entire pipeline was designed to be reproducible and computationally efficient, enabling model training on standard workstation environments.

Statistical Software and Metrics. All model training, evaluation, and statistical analysis were performed using the Python programming language (version 3.9+). Key libraries included Scikit-learn, XGBoost, and LightGBM [10]. The models were evaluated using metrics appropriate for imbalanced classification tasks: the Area Under the ROC Curve (ROC-AUC) and the Area Under the Precision-Recall Curve (PR-AUC), with PR-AUC providing a more robust measure of performance on rare classes [11]. Statistical comparisons between the best ensemble model and baseline models were conducted using standard ROC-AUC comparison procedures, with the resulting P-values reported in the text.

Listing 1. XGBoost model initialization with optimized parameters (Python)

The following Python fragment illustrates the core logic for initializing the XGBoost classifier with optimized parameters, including the metric selection for imbalanced data and the early stopping mechanism.

```
# Model Initialization (Python 3.9+)
import xgboost as xgb
from sklearn.model_selection import train_test_split

# Optimized parameters obtained via Bayesian Search
params = {
    'objective': 'binary:logistic',
    'learning_rate': 0.045,
    'n_estimators': 2000,
    'max_depth': 6,
    'subsample': 0.8,
    'eval_metric': 'aucpr',     # Metric optimized for imbalance
    'random_state': 42          # Ensures reproducibility
}

clf = xgb.XGBClassifier(**params)

# Assuming X_train, y_train, X_val, y_val are prepared datasets
# clf.fit(
#     X_train, y_train,
#     early_stopping_rounds=50,   # Early stopping prevents overfitting
#     eval_set=[(X_val, y_val)],
#     verbose=False
# )
```

Reproducibility and FAIR Compliance

To ensure methodological transparency and facilitate independent replication, the proposed pipeline adheres to the FAIR data principles (Findable, Accessible, Interoperable, and Reusable). All datasets used in this study originate from publicly accessible repositories, primarily ClinVar and gnomAD, which provide standardized variant annotation formats and stable identifiers. The computational workflow was implemented using open-source Python libraries, including Scikit-learn (version 1.3+), XGBoost (version 1.7+), and LightGBM (version 4.0+), ensuring long-term software reproducibility. Random seeds were fixed across all steps of training, validation, and evaluation to guarantee deterministic behavior.

Moreover, the feature-engineering and preprocessing procedures were scripted in a fully automated pipeline, enabling seamless reconstruction of the experimental environment on standard workstation hardware. This compliance with reproducible research practices strengthens the reliability of the findings and supports further integration of the proposed methods into clinical and research settings.

RESULTS AND DISCUSSION

Comparative Performance Analysis

Experimental validation confirmed the robustness and efficacy of the proposed ensemble models. The GBDT-based classifiers consistently outperformed baseline methods (such as Logistic Regression and SVM) in distinguishing between pathogenic and benign variants [12]. All models were evaluated on an independent test subset of 8,535 variants (approximately 15% of the total dataset), preserving the original class proportions. This ensured that the reported performance metrics reliably reflect model behavior under real-world class imbalance conditions.

The synergy between systematically engineered genomic features and boosting algorithms resulted in substantial improvements in predictive performance. The combined approach achieved higher AUC and precision–recall values compared to traditional single-model methodologies. High AUC values indicate strong discriminatory capacity, while improved precision–recall performance confirms robustness in handling rare pathogenic variants [4].

As illustrated in **Fig. 2,** the ROC curves of the ensemble models are consistently positioned closer to the top-left corner of the plot, demonstrating superior sensitivity–specificity trade-offs relative to the baseline classifier. Notably, the best-performing model (XGBoost) [5] achieved an AUC of 0.985 and a PR-AUC of 0.892 ($P < 0.001$ compared to SVM). These results confirm a statistically significant improvement over conventional baseline machine learning approaches.

The observed performance gains can be attributed to the ability of gradient boosting algorithms to capture complex nonlinear interactions within high-dimensional genomic feature spaces. Feature space expansion combined with ensemble optimization enhanced model generalization and robustness across datasets [13]. An additional performance increase was observed after extending the feature space with heterogeneous genomic descriptors, including conservation-related and sequence-context annotations. Compared to the minimal feature subset, the expanded representation improved model stability and reduced sensitivity to class imbalance, particularly for rare pathogenic variants. These findings suggest that biologically informed feature engineering contributes substantially to predictive robustness and complements the discriminative capacity of ensemble learning algorithms.

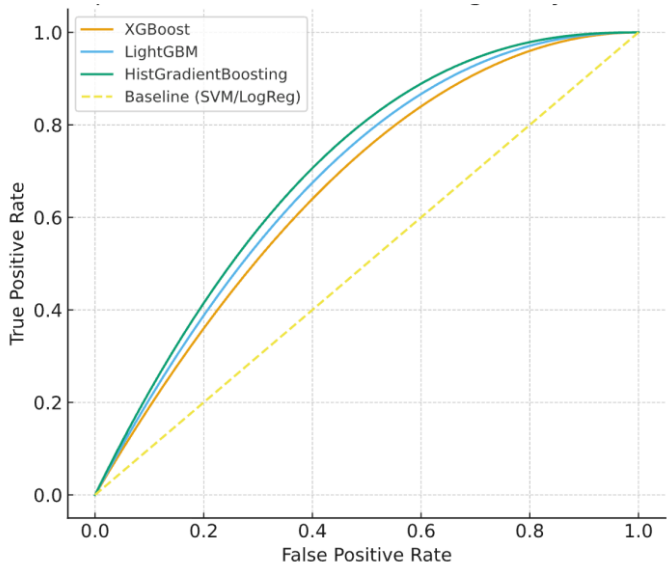

Fig. 2. Comparative Receiver Operating Characteristic (ROC) Curves for Pathogenicity Prediction

Comparative Receiver Operating Characteristic (ROC) Curves illustrating the predictive performance of the proposed ensemble models (HistGradientBoosting, XGBoost, LightGBM) versus a standard baseline classifier (e.g., SVM) on the ClinVar dataset. The ensemble approaches consistently exhibit higher Area Under the Curve (AUC) scores, demonstrating enhanced discriminatory power between pathogenic and benign variants

To further assess predictive quality under class imbalance, Precision–Recall (PR) curves were constructed for all classifiers (**Fig. 3**). Unlike ROC analysis, PR curves provide a more informative evaluation when the positive class is relatively rare. The ensemble models maintained consistently higher precision across a broad recall range compared to the baseline classifier, confirming their superior capability to identify pathogenic variants while limiting false positives. This behavior highlights the clinical relevance and practical applicability of gradient boosting frameworks in genomic variant prioritization.

Comparative Precision–Recall curves illustrating the performance of the proposed ensemble models (HistGradientBoosting, XGBoost, and LightGBM) in comparison with a baseline classifier (SVM) on the ClinVar test subset. The ensemble approaches demonstrate consistently higher precision across a wide range of recall values, confirming their robustness in handling class imbalance and rare pathogenic variant prediction.

Collectively, the ROC and PR analyses provide complementary evidence of the superior generalization capacity and clinical relevance of the proposed ensemble framework. Together, these findings establish the quantitative foundation for further model interpretability analysis.

Unlike previous studies that primarily focus on the predictive performance of individual machine learning algorithms, this work demonstrates that systematic biologically informed feature engineering combined with a lightweight reproducible pipeline improves model robustness and generalization while remaining computationally feasible on standard workstation environments.

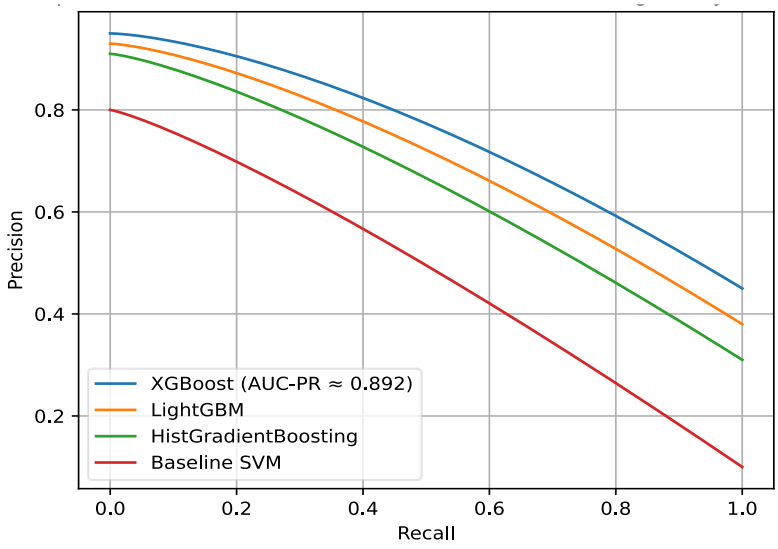

Fig. 3. Comparative Precision–Recall (PR) Curves for Pathogenicity Prediction

Feature Importance and Model Interpretability

A crucial component of this framework is the integration of Explainable Artificial Intelligence (XAI) principles. By utilizing SHAP-based interpretability methods, the model provides transparency regarding the contribution of specific biological features to individual predictions [14].

Analysis of feature importance across the ensemble models revealed that Allele Frequency (reflecting variant rarity in populations) and Evolutionary Conservation Scores were consistently ranked as the most influential features for predicting pathogenicity [3, 12]. This finding confirms the biological plausibility of the model's decisions, which is essential for ensuring clinical reliability. The lightweight and reproducible design of the pipeline, allowing training on standard workstations, makes the high-accuracy prediction methodologies widely accessible to the scientific community. Recent advances in deep learning for genomics have highlighted the growing importance of representation-learning approaches for capturing complex biological relationships that may not be explicitly encoded through handcrafted features [15].

Limitations and Future Considerations

Although the proposed ensemble framework demonstrates strong predictive capacity, several limitations must be acknowledged to contextualize the results. First, ClinVar annotations inherently reflect reporting and submission biases, with pathogenic variants more frequently documented in clinical populations from North America and Europe. This may introduce population stratification effects not fully captured by the models.

Second, the present study does not incorporate structural variants, splice-region effects, or long-range regulatory interactions, which increasingly contribute to the understanding of rare disease mechanisms and have been successfully modeled using deep learning approaches for noncoding variant interpretation [16]. While the engineered feature set captures essential population and conservation signals, additional molecular-level features – such as protein stability scores or deep-learning–derived sequence embeddings – may further enhance performance.

Finally, although gradient boosting provides a favorable balance between interpretability and accuracy, deep learning approaches such as graph neural networks or transformer-based genomic architectures may offer complementary perspectives. Future pipelines may benefit from hybrid or stacked-ensemble designs integrating both model families while maintaining explainability requirements essential for clinical adoption.

CONCLUSION

Ensemble learning techniques, particularly those based on advanced gradient boosting frameworks (HistGradientBoosting, XGBoost, and LightGBM), offer an effective, robust, and interpretable strategy for genomic variant classification and pathogenicity prediction.

The work successfully developed a computationally efficient and reproducible pipeline trained on systematically engineered and expanded feature representations derived from the open-access ClinVar repository. The proposed computational pipeline demonstrated that the systematic integration of biologically informed feature engineering with gradient boosting ensemble algorithms improves predictive robustness and generalization across heterogeneous genomic datasets. The expanded biologically informed feature space contributed not only to higher classification accuracy but also to improved predictive stability and greater resilience to class imbalance under real-world clinical genomic data conditions. This pipeline can be readily integrated into automated genomic analysis workflows and provides a practical foundation for future extensions involving additional biological data modalities and deep learning–based approaches.

The principal contribution of this study lies in the design and evaluation of a lightweight, reproducible computational pipeline that combines biologically informed genomic feature engineering, automated Bayesian hyperparameter optimization, and interpretable ensemble learning for genomic variant pathogenicity prediction.

Future work will focus on three main directions to further enhance predictive capacity: expanding the feature set through the integration of additional biological modalities (e.g., transcriptomic profiles), incorporating advanced deep learning components (e.g., neural embeddings) to capture increasingly complex relationships, and continuously optimizing hyperparameters and model architecture using automated search techniques for scalability on ever-growing large-scale genomic datasets. These continuous improvements are vital for translating complex genomic data into actionable clinical understanding.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, authorship, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [A.S.]; methodology, [A.S.]; validation, [A.S.]; writing – original draft preparation, [A.S.]; writing – review and editing, [A.S., I.K.]; supervision, [I.K.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] The 1000 Genomes Project Consortium. (2015). A global reference for human genetic variation. Nature, 526(7571), 68–74. https://doi.org/10.1038/nature15393

[2] Libbrecht, M. W., & Noble, W. S. (2015). Machine learning applications in genetics and genomics. Nature Reviews Genetics, 16(6), 321–332. https://doi.org/10.1038/nrg3920

[3] Kircher, M., Witten, D. M., Jain, P., O'Roak, B. J., Cooper, G. M., & Shendure, J. (2014). A general framework for estimating the relative pathogenicity of human genetic variants. Nature Genetics, 46, 310–315. https://doi.org/10.1038/ng.2892

[4] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

[5] Jain, A., et al. (2021). Classification of genetic variants using machine learning. arXiv preprint arXiv:2112.05154. https://arxiv.org/abs/2112.05154

[6] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953

[7] Quang, D., & Xie, X. (2015). DANN: A deep learning approach for annotating the pathogenicity of genetic variants. Genome Research, 25(10), 1554–1563. https://doi.org/10.1101/gr.184473.114

[8] Avsec, Ž., Agarwal, V., Visentin, D., Ledsam, J. R., Grabska-Barwinska, A., Taylor, K. R., Assael, Y., Jumper, J., Kohli, P., & Kelley, D. R. (2021). Effective gene expression prediction from sequence by integrating long-range interactions. Nature Methods, 18, 1196–1207. https://doi.org/10.1038/s41592-021-01252-x

[9] Kelley, D. R., Snoek, J., & Rinn, J. L. (2016). Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Research*, *26*(7), 990–999. https://doi.org/10.1101/gr.200535.115

[10] Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, *25*. https://doi.org/10.48550/arXiv.1206.2944

[11] Rentzsch, P., Schubach, M., Shendure, J., & Kircher, M. (2021). CADD-Splice – improving genome-wide variant effect prediction using deep learning-derived splice scores. *Genome Medicine*, *13*, 31. https://doi.org/10.1186/s13073-021-00835-9

[12] Ng, P. C., & Henikoff, S. (2003). SIFT: Predicting amino acid changes that affect protein function. *Nucleic Acids Research*, *31*(13), 3812–3814. https://doi.org/10.1093/nar/gkg509

[13] Alipanahi, B., Delong, A., Weirauch, M. T., & Frey, B. J. (2015). Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature*

Biotechnology, *33*(8), 831–838. https://doi.org/10.1038/nbt.3300
[14] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30. https://doi.org/10.48550/arXiv.1705.07874
[15] Zhou, J., & Troyanskaya, O. G. (2015). Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods*, *12*(10), 931–934. https://doi.org/10.1038/nmeth.3547
[16] Eraslan, G., Avsec, Ž., Gagneur, J., & Theis, F. J. (2019). Deep learning: new computational modelling techniques for genomics. *Nature Reviews Genetics*, *20*(7), 389–403. https://doi.org/10.1038/s41576-019-0122-6

АНСАМБЛЕВІ МЕТОДИ МАШИННОГО НАВЧАННЯ ДЛЯ ПРОГНОЗУВАННЯ ГЕНОМНИХ ВАРІАНТІВ

Андрій Смоляк, Іван Куньо
Кафедра оптоелектроніки та інформаційних технологій,
Львівський національний університет імені Івана Франка,
вул. Ген. Тарнавського, 107, Львів, 79017, Україна

АНОТАЦІЯ

Вступ. Стрімкий розвиток технологій секвенування нового покоління (СНП) призвів до накопичення величезних обсягів геномних даних, що відкриває нові можливості та виклики у виявленні генетичних варіантів, пов'язаних із захворюваннями. Традиційні статистичні або евристичні підходи часто не здатні врахувати складні, нелінійні взаємозв'язки між геномними ознаками та патогенністю варіантів. Інтеграція *ансамблевих методів машинного навчання* є перспективним напрямом підвищення точності та інтерпретованості результатів у геномних дослідженнях. У цій роботі розроблено ансамблеву модель для класифікації генетичних варіантів із використанням алгоритмів градієнтного підсилювання, натренованих на даних бази ClinVar.

Матеріали та методи. Для навчання використано відкритий набір даних бази ClinVar, що містить анотовані генетичні варіанти з позначеннями клінічного значення. Після етапів попередньої обробки та інженерії ознак застосовано ансамблеві алгоритми *HistGradientBoosting*, *XGBoost* та *LightGBM*. Було виділено ключові прогностичні ознаки, такі як частота алелей, тип варіанту, співвідношення транзицій/трансверсій тощо. Запропонований конвеєр є відтворюваним та оптимізованим для роботи на стандартних робочих станціях.

Результати. Проведені експерименти підтвердили ефективність запропонованого підходу для класифікації патогенних та доброякісних генетичних варіантів. Отримані результати свідчать, що інтеграція біологічно обґрунтованих геномних ознак, зокрема частоти алелей, функціональних анотацій, показників еволюційної консервативності та характеристик послідовностей, сприяє підвищенню прогностичної точності, стійкості та узагальнювальної здатності моделей на гетерогенних геномних наборах даних. Поєднання розширеного простору ознак із ансамблевими алгоритмами градієнтного підсилювання забезпечило високі значення ROC-AUC та PR-AUC на незалежній тестовій вибірці при збереженні обчислювальної ефективності та відтворюваності.

Висновки. Ансамблеве машинне навчання є ефективним та інтерпретованим підходом для класифікації генетичних варіантів і прогнозування їх патогенності. Проведене дослідження показало, що поєднання біологічно обґрунтованої інженерії ознак із ансамблевими моделями градієнтного підсилювання сприяє підвищенню прогностичної стійкості, узагальнювальної здатності та стійкості до дисбалансу класів у гетерогенних геномних наборах даних. Запропонований відтворюваний обчислювальний фреймворк, побудований на основі анотацій ClinVar та розширеного простору геномних ознак, є обчислювально ефективним рішенням і має потенціал для інтеграції в автоматизовані системи аналізу геномних даних. Подальші дослідження будуть спрямовані на інтеграцію додаткових біологічних модальностей даних, компонентів глибокого навчання та подальшу оптимізацію моделей для роботи з великомасштабними геномними вибірками.

Ключові слова: ансамблеве навчання, прогнозування геномних варіантів, патогенні мутації, градієнтне підсилювання, інженерія ознак, біоінформатика.

Received / Одержано	Revision / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
09 March, 2026	26 June, 2026	26 June, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 145–164
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.728.3:004.85

ADAPTIVE PROTOCOL SELECTION LAYER FOR CONTEXT-AWARE MULTI-PROTOCOL DATA DELIVERY IN CLOUD–EDGE SYSTEMS: A CONTEXTUAL-BANDIT FOUNDATION WITH BAYESIAN WARM-START

Roman Nazarevych* , ***Ivan Bolesta***
Department of Radiophysics and Computer Technologies,
Ivan Franko National University of Lviv
107 Gen. Tarnavsky Str., UA-79017 Lviv, Ukraine

Nazarevych, R., & Bolesta, I. (2026). Adaptive Protocol Selection Layer for Context-Aware Multi-Protocol Data Delivery in Cloud–Edge Systems: A Contextual-Bandit Foundation with Bayesian Warm-Start. *Electronics and Information Technologies*, 34, 145–164. https://doi.org/10.30970/eli.34.9

ABSTRACT

Background. Protocol performance in cloud–edge systems is regime-dependent: the transport that minimizes latency, loss, or CPU cost changes with payload, rate, link type, and signal quality, and tail latency often inverts the median ranking. Static choices and hand-tuned heuristic switchers cannot track these shifts, and many nearby adaptive-selection prototypes remain deterministic scorers, not online learners. This paper establishes the theoretical foundation for an adaptive protocol selection layer that treats protocol choice at decision ticks as online decision-making under partial feedback, demonstrated on isolated proof-of-concept examples.

Materials and Methods. Protocol selection is formulated as a contextual multi-armed bandit over the five supported implementation transports and solved with LinUCB: a per-arm ridge-regression estimator with an upper-confidence-bound exploration term, computed by Gaussian elimination. The four-dimensional context couples a log-linear payload feature, normalized Wi-Fi signal and link-rate features, and a bias term. Bounded penalty and peer-relative rewards are derived from blended median/tail latency, goodput, and loss. Non-stationarity is handled by per-tick geometric decay, and a Bayesian warm-start encodes offline ridge priors in the initial state; normal per-tick decay may scale that prior before the first logged score.

Results and Discussion. The five-protocol formulation is demonstrated on isolated Raspberry Pi 5B and Pi Zero 2 W examples; the adaptive algorithm uses five protocols (gRPC, gRPC-bidi, MQTT, CoAP, HTTP). Baseline runs confirm the best transport changes with payload, host, and objective; decision logs show LinUCB concentrating on a strong context-conditioned arm in stable regimes, while a deliberately retained cold-start miss exposes early-evidence path dependence; a Bayesian warm-start improves cold-start latency and throughput.

Conclusion. The paper gives a reproducible contextual-bandit formulation with Bayesian warm-start for multi-protocol cloud–edge data delivery. It defines context, reward, and exploration sufficient for further research.

Keywords: contextual bandit, LinUCB, adaptive protocol selection, cloud–edge systems, Bayesian prior, exploration-exploitation.

INTRODUCTION

Cloud–edge systems, including remote-instrumentation deployments such as Internet-accessible laboratories [1], deliver data over a few interchangeable application transports – gRPC [2], MQTT [3], CoAP [4], HTTP – each with different framing, session, congestion, and reliability semantics, and each evolving independently (QUIC and HTTP/3 [5, 6] are recent instances). Across cross-protocol IoT benchmarks [7], no single transport dominates across regimes: the protocol that minimizes median round-trip time at a 256-byte payload over Wi-Fi is often not the one that minimizes tail latency, loss, or edge CPU at a larger payload or on a wired link, and the ranking can invert between the 50th and 99th percentiles. The "best" protocol is thus a function of context, not a constant.

The conventional response is either a static, globally fixed protocol choice or a hand-tuned heuristic switcher with weighted scores, hysteresis, and moving averages. Both are brittle: a static choice cannot follow regime changes, and a heuristic switcher encodes its trade-offs in fixed weights and thresholds that must be re-tuned per deployment and gives no principled treatment of the exploration/exploitation dilemma – choosing the apparently best protocol now forecloses the information needed to tell whether an under-measured protocol is better. The closest prototypes in the literature remain deterministic scorers; online-learning theory, by contrast, frames this problem as regret minimization under partial (bandit) feedback [8, 9, 10, 11, 12, 13].

This paper develops the theoretical foundation for an adaptive protocol selection layer that closes this gap. We formulate protocol choice at adaptive decision ticks as a contextual multi-armed bandit and instantiate it with LinUCB [14, 15], whose decision index decomposes transparently into exploitation and uncertainty-driven exploration terms. We also derive a Bayesian warm-start that converts an offline benchmark corpus into per-arm priors, shortening the cold-start phase in which an unprimed bandit selects almost randomly. The scope is deliberate: we present the context design, reward function, exploitation/exploration mechanism, non-stationarity treatment, and prior warm-start – all reflecting a working reference implementation – and demonstrate them on isolated worked examples from a heterogeneous Raspberry Pi testbed. The systematic payload/rate evaluation, offline-replay parameter study, and reward-shaping and exploration-coefficient ablations are deferred to a companion paper; this article fixes the methodological baseline for that study.

This paper's contributions are: (i) a precise contextual-bandit formulation of multi-protocol cloud–edge data delivery with an explicitly bounded context vector; (ii) two bounded reward shapings – an absolute payload-aware penalty and a peer-relative advantage – derived from latency, loss, and goodput; (iii) a decay-based non-stationarity model integrated into the per-arm ridge estimator; (iv) an algebraic Bayesian warm-start identity with a tunable prior strength; and (v) a proof-of-concept demonstration on a heterogeneous Raspberry Pi testbed that makes the exploitation/exploration split, the safety filter, and the warm-start identity observable on real traffic. All equations map directly to the implemented controller, so the formulation is independently reproducible.

MATERIALS AND METHODS

Problem formulation

Define the action set A as the five supported transports: $A = \{gRPC, gRPC\text{-}bidi, MQTT, CoAP, HTTP\}$. At each decision tick t, the controller observes a context vector $x_t \in \mathbb{R}^4$ (defined below), selects one protocol $a_t \in A$, routes subsequent messages through it for the configured decision interval, and later observes a scalar reward r_t derived

from the observed latency, loss, and throughput of that protocol over the interval since the previous tick. Crucially, the controller observes a reward only for the protocol it selected (and for any protocol that produced resolved traffic in the window) – this is the defining partial-feedback structure of a bandit problem [9, 13]. Conceptually, this is a regret-minimization problem against the unknown context-conditioned optimal protocol rather than the maximization of a fixed weighted score, which is what motivates the bandit formulation adopted here. A quantitative regret analysis – for example, cumulative-regret curves on offline replay – is beyond the scope of this foundational paper and is deferred to the companion paper; the worked examples below instead characterize behavior through aggregate latency and throughput statistics and arm-selection shares. Decisions are pinned for a configurable interval after a switch to prevent flapping. Rewards are accumulated from an append-cursor delta so that each update reflects only the samples produced since that arm's previous update.

Context vector

The per-decision context is a four-dimensional vector that combines a workload feature, two link-quality features, and a bias term:

$$x = [payloadNorm, rssiNorm, txBitrateNorm, 1]^T. \quad (1)$$

The payload feature maps message size log-linearly onto the unit interval over the range [256 B, 1 MB] and saturates outside it (with $LN2 = \ln 2$ and the constant $12 = \log_2(1\ \text{MB} / 256\ \text{B})$):

$$payloadNorm(p) = clip\left\{\frac{\ln\left\lceil \frac{\max(p,1)}{256} \right\rceil}{LN2 \cdot 12}, 0, 1\right\}. \quad (2)$$

A logarithmic scale matches how transports experience size – a 256 B → 1 KB increase is perceptually similar to a 256 KB → 1 MB increase – and, more importantly, the explicit clip keeps the context norm $||x||$ bounded. Since the LinUCB exploration term grows with $||x||$, an unbounded payload feature would cause the exploration bonus to diverge at large payloads, and the bandit would fail to converge. Bounding the feature is done to avoid this problem. The link-quality features are sourced from a background poller of the edge device's network endpoint and normalized as:

$$rssiNorm = \text{clip}\left\{\frac{[rssi_{dBm} - (-90)]}{60}, 0, 1\right\}, \ default\ 0.5, \quad (3)$$

$$txBitrateNorm = \text{clip}\left(\frac{txMbps}{300}, 0, 1\right), \ default\ 0.5 \quad (4)$$

RSSI is mapped from [−90, −30] dBm and link rate from [0, 300] Mbps (a typical 802.11n peak; 300 Mbps is the two-stream 40 MHz 802.11n maximum). On non-WiFi links, or before the first poll completes, both default to the neutral value 0.5. The constant bias term absorbs each arm's intercept. Payloads are additionally discretized into a small number of buckets (default 8) using the same *payloadNorm* mapping; the buckets index the empirical baselines used by the reward function. We note a known degeneracy: on a

wired link *rssiNorm* and *txBitrateNorm* are held at the neutral constant 0.5, so the effective feature subspace collapses from four dimensions to two: *[payloadNorm, bias].* This does not cause underdetermination in the numerical sense – the precision matrix A_a is initialized to the identity (the ridge regularizer) and receives positive diagonal mass at every update, so the linear system $A_a z = b_a$ stays full-rank and uniquely solvable throughout. The practical consequence is that the two constant columns add no arm-differentiating signal: on a wired link, per-arm ordering reflects only payload sensitivity and the per-arm intercept, while any wired-link quality variation (interface speed, congestion, half-duplex state) stays invisible to the learner. Two of the four model degrees of freedom are therefore wasted on informationally dark dimensions. The exploration bonus $\alpha\sqrt{x^T A_a^{-1} x}$ does still converge as *payloadNorm* varies across ticks, but convergence is slower than in a full-rank context because the effective information per tick is lower. This limitation motivates the richer wired-context design planned for the companion paper.

LinUCB estimator: exploitation and exploration

Each arm *a* maintains a ridge-regression state: a 4×4 precision matrix A_a and a 4-vector b_a. A_a is the accumulated outer-product design matrix (initialized to the identity, which is the ridge regularizer [16]); b_a is the reward-weighted feature accumulator. The point estimate of the per-arm weight vector is the ridge solution:

$$\theta_a \;=\; A_a^{-1}\, b_a. \tag{5}$$

The inverse is never materialized; the linear system $A_a z = rhs$ is solved by Gaussian elimination with partial pivoting [17], an $O(d^3)$ operation that is negligible for $d = 4$ and numerically stable for the well-conditioned, identity-regularized A_a. Each protocol is scored by its upper-confidence-bound index, which is the sum of an exploitation term and an exploration term:

$$p_a(x) \;=\; \theta_a^T\, x + \alpha \cdot \sqrt{x^T\, A_a^{-1}\, x} \tag{6}$$

The first term $\theta_a^T x$ is the model's current best estimate of the expected reward for context x (pure exploitation). The second term is the one-sided confidence width: $x^T A_a^{-1} x$ is large when arm *a* has few observations in the region of feature space near x, so the bonus rewards uncertainty. The coefficient α (default 1.0) sets the exploration intensity; $\alpha \to 0$ is greedy exploitation, larger α explores more aggressively [14, 18]. At each decision tick, the candidate set is first reduced to the protocols that satisfy the viability filter (described below), which excludes any arm that is mechanically unsuitable for the current context – for example, CoAP above its supported payload boundary, or an arm currently in viability cooldown. The controller then evaluates the upper-confidence-bound index $p_a(x)$ for each remaining arm and selects the one that maximizes it:

$$a_t = \underset{a \,\in\, viable}{\operatorname{argmax}}\; p_a(x_t). \tag{7}$$

At cold start $\theta_a = 0$, for every arm, the index reduces to the exploration term alone; at the very first tick $A_a = I$, for all arms, and the shared context makes the bonus $\alpha\sqrt{x^T x}$. Identical across arms, so the initial pick is an arbitrary tie-break, not real exploration. Once arms begin to differ, the bonus favors whichever arm is least sampled near x, so the

controller explores; as evidence accumulates, the exploitation term dominates and the policy concentrates on the context-conditioned best arm. This is the principled treatment of the exploration/exploitation trade-off that fixed heuristic scorers lack.

Online update and non-stationarity

After the reward r for context x is computed for an arm, its state is updated by the standard recursive least-squares step:

$$A_a \leftarrow A_a + x \cdot x^T, \qquad b_a \leftarrow b_a + r \cdot x. \tag{8}$$

Network conditions drift, so old evidence must not dominate indefinitely. Once per active decision tick (not on pinned ticks, and not for an arm in viability cooldown), every arm is decayed geometrically toward the regularized:

$$A_a \leftarrow \gamma A_a + (1-\gamma)I, \qquad b_a \leftarrow \gamma b_a \tag{9}$$

with $\gamma \in (0, 1]$ (default 0.99). The $(1-\gamma)I$ term prevents A_a from becoming near-singular as the old signal fades, keeping the solve well-conditioned [19]. Values of γ near 1 give long memory; smaller γ adapts faster but is noisier. For arms placed in cooldown by the viability filter, the geometric decay step (Eq. 9) is suspended for the entire quarantine period. Without this exception, every active tick would shrink both A_a and b_a toward the identity prior, gradually erasing the large negative reward that triggered the gate in the first place. As θ_a relaxes back toward zero and $x^T A_a^{-1} x$ grows, the UCB index for the quarantined arm would rise purely as an artefact of forgetting, and the controller would re-select it as soon as cooldown expires – only to have it gated again. Freezing decay during cooldown preserves the evidence that justified the gate, so the arm is re-admitted based on explicit re-evaluation rather than uncertainty.

Reward function

The LinUCB selector learns from a scalar reward derived from three performance indicators – latency, message loss, and throughput – designed to be both interpretable and sufficiently discriminative across workload conditions. The simplest form is an absolute reward that compares each measured outcome against fixed reference values: latency is penalized when it exceeds a predefined budget, throughput when it falls below a target, and loss as its rate increases. This directly measures how far a protocol deviates from its expected performance targets.

However, fixed reference values are not robust across payload sizes: larger messages naturally require more transmission and processing time, so a single latency budget applied to all payloads is frequently reached or exceeded by larger ones. Because the penalty is clipped to the interval [0, 1], clearly different outcomes then collapse to the same saturated value – under a fixed 20 ms budget, latencies of 35, 60, and 120 ms all receive the maximum penalty. This saturation reduces the resolution of the reward signal, giving the bandit nearly identical feedback for protocols whose actual performance differs substantially and weakening learning in large-payload regimes. To address this, we use two reward-shaping modes: absolute and relative.

For each arm for which at least one resolved response is recorded within the delta window, an observation is formed from the windowed median (p_{50}) and tail (p_{99}) round-trip times, payload size, and loss count. The *latency signal* ℓ is defined as a fixed equal-weight combination of the two percentiles, which makes tail behavior one of the primary objectives:

$$\ell = 0.5 \cdot p_{50} + 0.5 \cdot p_{99}, \tag{10}$$

$$g = \frac{payload \cdot 1000}{\max(\ell, 1)}. \tag{11}$$

Here g is an internal goodput proxy in bytes per second, used only to rank arms inside the reward; it is distinct from the message-rate throughput (messages per second) reported in the Results. If an arm sent messages but received no responses in the window, its reward is fixed at −1 (the worst value) in both reward modes. Otherwise, one of two bounded shapings is applied.

Absolute reward mode converts a single protocol observation into a score between −1 and 0, where 0 means the protocol performed perfectly and −1 means it performed as badly as possible. The score is a penalty – the worse the protocol behaved, the more negative it gets.

To compute this penalty, the system first sets a target budget for each metric – essentially an acceptable performance threshold. For latency, the budget B_ℓ (Eq. 12) is either:

- *a learned threshold*: the protocol's recent average latency (tracked as an EWMA – a smoothed running average) multiplied by a headroom factor of 1.25, giving the protocol 25% slack above its typical performance, or
- *a cold-start fallback*: an analytic formula based on a fixed base of 15 ms plus up to 135 ms scaled by normalized payload size, used when not enough data has been collected yet.

From the observation, three individual penalty components are computed (Eq. 13):

- $latPen$ – how much the measured latency ℓ exceeds the budget B_ℓ, clipped to [0, 1]. Zero if latency is within budget; 1 if it's at or beyond the budget.
- $lossPen$ – packet loss rate q scaled by 10, clipped to [0, 1]. A loss rate of 10% or more fully saturates this component (score = 1), meaning the packet loss has a very high impact on reward.
- $thrPen$ – how far the goodput proxy g falls short of its budget B_g, clipped to [0, 1]. Zero if throughput meets the target or 1 if it falls to zero.

These three components are combined into a single reward (Eq. 14) using fixed weights: latency weight $w\ell = 0.45$, loss weight $w_q = 0.35$, goodput weight $w_g = 0.20$. Latency and loss together account for 80% of the score, reflecting that they matter more than raw throughput on constrained edge links. The division by the sum of weights is technically an identity at the defaults (they already sum to 1), but is kept explicitly so the reward stays bounded if we decide to adjust the weights.

$$B_\ell = baseMs + scaleMs \cdot payloadNorm, \;\; or \;\; B_\ell = ewma_\ell \cdot headroom, \tag{12}$$

$$\begin{aligned} latPen &= \mathrm{clip}\left(\frac{\ell}{B_\ell}, 0, 1\right); \\ lossPen &= \mathrm{clip}(10\, q, 0, 1); \\ thrPen &= \mathrm{clip}\left(1 - \frac{g}{B_g}, 0, 1\right), \end{aligned} \tag{13}$$

$$r_{abs} = -\text{clip}\left(\frac{w_\ell \cdot latPen + w_q \cdot lossPen + w_g \cdot thrPen}{w_\ell + w_q + w_g},\ 0,\ 1\right). \quad (14)$$

The *EWMA* (Exponentially Weighted Moving Average) baselines used for the budgets are updated each round using (Eq. 15) with smoothing factor $\beta = 0.20$ – meaning each new observation contributes 20% to the running estimate and the previous average retains 80%.

$$m \leftarrow (1-\beta)\cdot m + \beta \cdot obs, \quad \beta = EWMA\ factor,\ default\ 0.20 \quad (15)$$

Relative reward mode uses the same observation and running averages, but instead of scoring against a predefined baseline, it scores the protocol relative to the other candidate protocols. This mode activates once at least two arms have accumulated enough observations to have stable EWMA baselines in the current payload bucket.

The comparison baselines – med_ℓ, med_g, med_q – are the medians of the ready arms EWMA values that were measured by now. They represent the typical performance level of the available protocols as seen so far.

Three advantage signals are computed (Eq. 16):

A_ℓ – log-ratio of the peer median latency to this arm's latency. Positive when this arm is faster than the median.

A_g – log-ratio of this arm's goodput proxy to the peer median. Positive when this arm has higher throughput.

A_q – linear difference between the peer median loss rate and this arm's loss rate, scaled by 10. Positive when this arm loses fewer packets.

These are combined with the same weights into a single advantage Δ (Eq. 17).

$$A_\ell = \ln\left(\frac{med_\ell}{\ell}\right);\ \ A_g = \ln\left(\frac{g}{med_g}\right);\ \ A_q = \left(med_q - q\right)\cdot 10, \quad (16)$$

$$\Delta = \frac{w_\ell \cdot A_\ell + w_q \cdot A_q + w_g \cdot A_g}{w_\ell + w_q + w_g}. \quad (17)$$

Because *Δ* is on an arbitrary scale, it is normalized (Eq. 18) by centering on the median advantage across all ready arms and dividing by a robust spread measure (MAD – median absolute deviation, floored at 0.02 to avoid division by near-zero). This makes the score comparable across different operating regimes.

$$c = \text{median}(\Delta_i);\ \ s = \max(\text{median}\ |\Delta_i - c|, minSpread);\ \ \widehat{\Delta} = \frac{\Delta - c}{s}. \quad (18)$$

The normalized advantage is then passed through a $\tanh$ function clipped to [−1, 1] (Eq. 19), with a gain of 2.0 that sharpens separation between protocols that are close in performance. Unlike the absolute reward (always ≤ 0), the relative reward can be positive; a protocol that outperforms its peers gets a positive score, which more aggressively steers the bandit toward the best option in a stable environment.

$$r_{rel} = \tanh\left(gain \cdot \hat{\Delta}\right) \quad (19)$$

Until two baselines are ready, the controller falls back to the absolute reward. The relative reward lies in [−1, 1] and can be positive when a protocol beats the current peer median, sharpening the separation between near-tied arms in a stable regime; because the baseline is online and payload-bucket-specific, relative reward values are comparable within a regime but not across unrelated regimes.

Bayesian prior warm-start

The LinUCB selector normally starts with no knowledge about which protocol is good. At the beginning of a run, all protocol arms have almost the same model parameters, so the selector must spend some time exploring all the protocols before it can learn which one performs best. This early exploration phase wastes time and may deliver poor performance in the first few runs. To reduce this cost, each protocol arm is pre-loaded with a starting model fitted from historical (non-adaptive) benchmark data. This technique – seeding a bandit from past observations – is a well-established way to shorten the exploration phase [20].

The pre-loading works as follows: for each protocol separately, a ridge regression model [16] is fitted on the historical data. Its target is an offline utility score, a single number that summarizes how well a protocol behaved in the past. This score combines three factors:

- Reliability – how often the protocol succeeded without errors
- Tail-latency sigmoid – a smooth penalty for high worst-case latency (using a sigmoid so the penalty grows gradually, not as a hard cutoff)
- CPU headroom – how much processing capacity was left unused

This offline score uses the same input features as the online model, but it is a different quantity from the online reward computed in equations (Eq. 14) and (Eq. 19) – it is only used to initialize the model, not during live operation. The target y in (Eq. 20) multiplies a reliability term $(1 - loss)$, a tail-latency sigmoid $\sigma(-p_{99}/\tau)$, and a CPU headroom term $(1 - cpu)$, and the model parameters θ are found by ridge regression with regularizer λ:

$$\min_\theta \sum_i (y_i - \theta^T x_i)^2 + \lambda\|\theta\|^2, \quad y = (1 - loss) \cdot \sigma\left(\frac{-p_{99}}{\tau}\right) \cdot (1 - cpu), \quad (20)$$

where σ is the logistic function, τ is an RTT decay scale in milliseconds, λ is the ridge regularizer, and loss and CPU are normalized fractions in [0, 1]. The prior JSON used at runtime stores fitted vectors θ_{prior} keyed by protocol, together with metadata such as d, τ_{ms}, and λ. At runtime, each arm is initialized so that its ridge state algebraically encodes the prior; the online controller loads the θ vectors and does not re-fit the offline prior. Setting the initial state to a scaled identity and a scaled prior:

$$A_a^{(0)} = s \cdot I\,, \qquad b_a^{(0)} = s \cdot \theta_{prior}. \quad (21)$$

Substituting the initial values from (Eq. 21) into the standard parameter update (Eq. 5) confirms that the model starts at exactly the prior vector:

$$\theta_a^{(0)} = (s \cdot I)^{-1} \left(s \cdot \theta_{prior}\right) = \theta_{prior}. \quad (22)$$

The initial setup encodes the prior knowledge directly into the model's starting state. Before any real data arrives, the model behaves as if it has already seen s observations consistent with the prior. The scalar s (default 1.0) controls how strongly the prior holds: a larger s means each new real observation has less relative influence, so the model takes longer to drift away from the prior. As more data accumulates, the prior's influence fades naturally.

One practical note: the per-tick decay step (which gradually reduces the weight of old observations) can run before the first real score is logged, so equations (Eq. 21) and (Eq. 22) describe the initialization state.

Finally, if a prior file is specified but fails to load, the system stops execution. This ensures a warm-start run is never accidentally treated as a cold one.

Safety and viability filter

Learning and reliability are kept separate. Before the bandit index (Eq. 6) is compared, a viability filter removes protocols that must not be selected in the current context, independently of their estimated reward: CoAP is hard-excluded for any payload strictly above a fixed 1900 B limit, an arm whose recent loss exceeds a threshold or whose recent receive count is zero (after a minimum sample count) is gated, and a gated arm enters a cooldown whose duration grows by exponential backoff on consecutive gate fires and resets after a healthy window. If every candidate is filtered out, the controller falls back to a configured safe-default protocol. This filter is a hard constraint applied to the action set before the reward calculation. Keeping the safety constraint outside the learned objective, rather than encoding it as a conservative reward penalty as in conservative-bandit formulations [21], means safety does not depend on the learner having converged.

Reference testbed

The formulation above is implemented in a working multi-protocol benchmark stack. The testbed consists of two edge devices – a Raspberry Pi 5 Model B (8 GB RAM, Gigabit Ethernet) and a Raspberry Pi Zero 2 W (512 MB RAM, 2.4 GHz Wi-Fi) – and a laptop/desktop client on Gigabit Ethernet. A deterministic offline replay harness drives the identical production selector against per-protocol isolated-baseline traces, enabling parameter studies without re-running on hardware. The selective worked examples reported below exercise this testbed only to make the formulation's moving parts observable on real traffic; the exhaustive payload/rate sweep, the replay parameter study, and the reward-shaping and exploration-coefficient sensitivity analyses are deferred to the companion paper. The software stack is implemented in Kotlin 1.9.22 (JVM 17). Transport client library versions: gRPC-Netty 1.62.2 and Protobuf 3.25.3 (gRPC and gRPC-bidi), HiveMQ MQTT Client 1.3.3 (MQTT), Eclipse Californium 3.9.0 (CoAP), and the Java built-in HTTP server from JDK 17 (HTTP).

Code and data availability

The selector implementation, the multi-protocol benchmark stack, and the decision logs used for the worked examples reported in this paper are publicly available at https://github.com/lemberh/proto-bench-pub/releases/tag/ELIT_2026Q2_publication. The repository provides the LinUCB selector, the per-protocol isolated-baseline traces, and the deterministic offline replay harness, enabling full reproduction of the reported results given the same configuration and candidate protocols. Also referenced version of the repository contains test results used in this paper and decision logs in the *bench-runner/results* directory.

RESULTS AND DISCUSSION

This paper is the foundational first part of a two-part series; its purpose is to establish the methodology and to demonstrate it on isolated worked examples. The reference testbed described above is exercised on the two heterogeneous edge devices – a Raspberry Pi 5 Model B on Gigabit Ethernet and a Raspberry Pi Zero 2 W on 2.4 GHz Wi-Fi – with a laptop client. Unless stated otherwise, the adaptive examples below use throughput mode with *throughputMaxInFlight* = 1 where applicable, LinUCB with *--banditAlpha* 0.5 (alpha=0.5), *--banditDecay* 0.99 *(decay factor* $\gamma = 0.99$*), --banditRewardMode relative*, and the five logged candidate protocols *{gRPC, gRPC-bidi, MQTT, CoAP, HTTP}*. The results below are deliberately selective: a baseline characterization that motivates the formulation, two decision-log walkthroughs that make the LinUCB index and the Bayesian warm-start observable, and an isolated-baseline reading of the selected runs. The exhaustive payload/rate sweep, the offline-replay parameter study, and the reward-shaping and α/γ sensitivity analyses are the subject of the companion paper and are not claimed here.

Baseline regime dependence

The isolated baseline results confirm that protocol performance is regime-dependent. Across the Raspberry Pi 5B and Raspberry Pi Zero 2W runs, the protocol minimizing median round-trip time is often not the protocol minimizing tail round-trip time, maximizing throughput, or minimizing edge CPU. The resulting winner grid (**Fig. 1**) changes across payloads and devices, which rules out a single globally optimal transport and motivates adaptive selection at decision ticks. These results confirm empirically what the Introduction asserts: optimal protocol choice is context-dependent, and a context-conditioned learner is therefore the correct instrument to select the optimal protocol, not a fixed selection or a statically weighted rule.

Safety gates are empirically justified

The separation between learning and reliability in the Materials and Methods section is not only a modeling choice; it is supported by the measurements. In our experimental setup, CoAP shows a sharp loss cliff when the payload approaches and exceeds the supported message-size region around 1900 B (**Fig. 2**). Below this boundary, CoAP remains competitive and can provide low-latency communication for small messages. Once the payload moves beyond the supported range, however, loss increases abruptly and can approach total failure.

This behavior should not be interpreted as a universal fixed limit of the CoAP protocol itself. The observed cliff is implementation- and platform-dependent. CoAP performance near large payloads is affected by the endpoint configuration and library behavior, including message-size limits, resource-body limits, block-wise transfer handling, UDP packet sizing and fragmentation, receive buffers, timeout behavior, and memory available on the edge device. These factors may differ across hardware platforms, operating systems, programming languages, and CoAP libraries.

The CoAP result illustrates why the selector must distinguish between learning which admissible protocol is best and detecting protocols that are no longer operationally safe. When a protocol enters a failure region caused by payload limits or resource constraints, continued exploration does not provide useful performance information. Instead, it mostly produces failed requests and distorted reward observations. A soft reward penalty alone is insufficient in this case: LinUCB may still occasionally re-sample a mechanically failing arm because exploration is part of its decision rule.

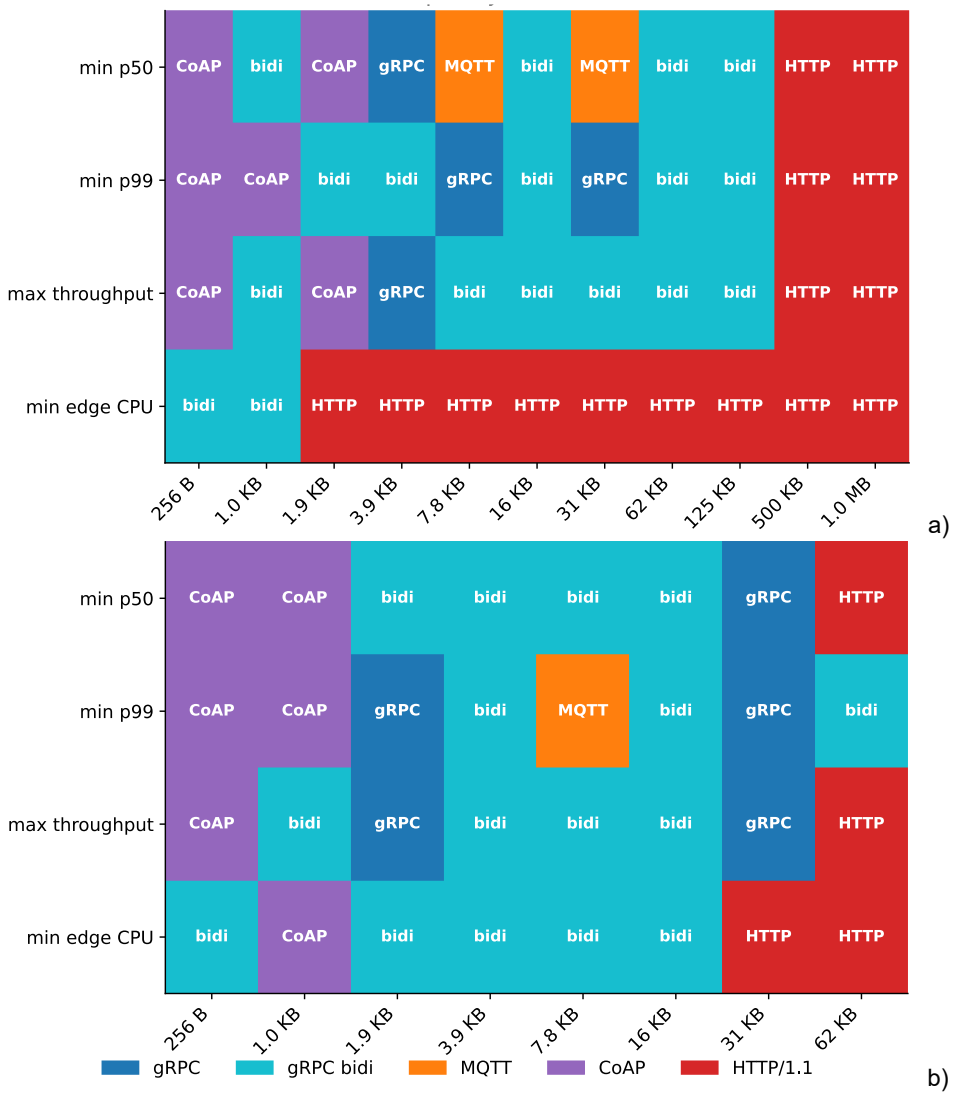

Fig. 1. Best-protocol grid by payload size and optimization objective (p_{50} RTT, p_{99} RTT, throughput, edge CPU): (a) Raspberry Pi 5B; (b) Raspberry Pi Zero 2 W. The winning transport changes across cells, so no single protocol dominates.

LinUCB convergence: a worked example

A clean convergence example is the Raspberry Pi Zero 2 W throughput run at a 256 B payload. Its decision log contains 282 scored UCB decision entries, excluding fallback/event records, with 51 switches; the selected-arm share concentrates on CoAP (≈78.7 %), with MQTT (≈9.6 %) and gRPC-bidi (≈8.9 %) absorbing most of the remainder. The aggregate outcome is p_{50} = 3.548 ms, p_{99} = 18.336 ms, throughput = 198.6 msg/s, and zero effective loss. **Fig. 3** plots the per-arm total UCB index over time. The controller begins in an exploration-dominated phase in which the indices are close and selection alternates, then the exploitation term $\theta^{\mathrm{T}}x$ separates the arms and selection concentrates on the context-conditioned best protocol. This is the exploitation/exploration decomposition of the UCB index made directly observable on real traffic, and it shows that the bandit

leaves its initial exploration phase and locks onto a strong arm for the observed context without external tuning.

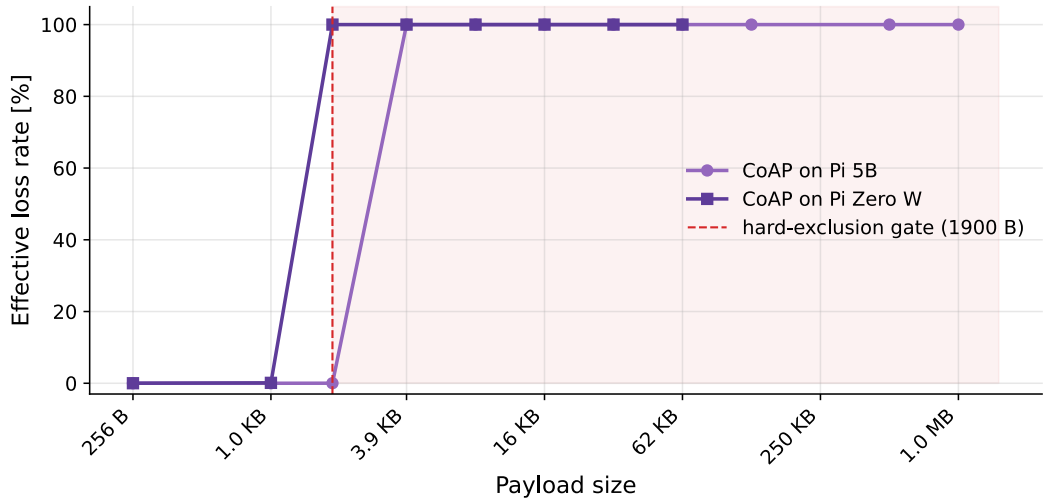

Fig. 2. CoAP message-loss rate versus payload size. The sharp drop just above the 1900 B supported message boundary motivates the hard CoAP payload exclusion (for payloads strictly above 1900 B) in the viability filter, independent of estimated reward.

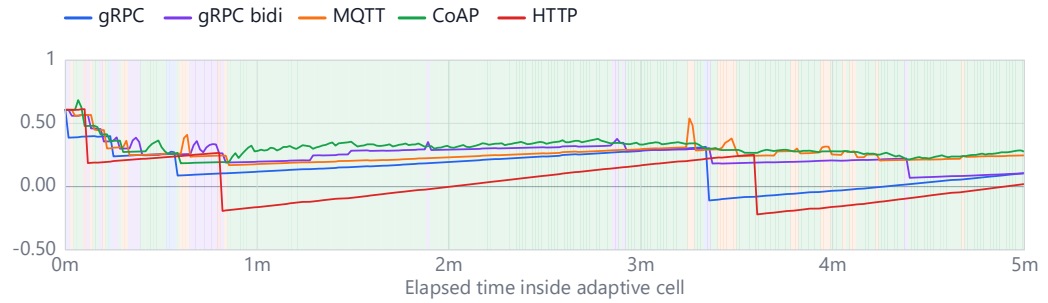

Fig. 3. Per-arm total UCB index over the decision sequence for the Raspberry Pi Zero 2 W 256 B throughput run (282 scored decisions, 51 switches). Each curve is the total index for one protocol arm; the controller begins in an exploration-dominated phase where indices are near-tied and selection alternates, after which the exploitation term ($\theta^{\mathrm{T}}x$) separates the arms and selection concentrates on CoAP (≈78.7% share; MQTT ≈9.6%, gRPC-bidi ≈8.9%). Aggregate outcome: p_{50} = 3.548 ms, p_{99} = 18.336 ms, throughput = 198.6 msg/s, zero effective loss.

Cold-start path dependence

The Pi 5B adaptive sweep illustrates the cold-start behavior of an online learner. At 1900 B, two cold runs followed different trajectories: run 1 concentrated on MQTT (75.3 % share) and missed the isolated-baseline envelope – p_{50} 3.271 ms, a +37.8 % p_{50} gap against the CoAP p_{50} reference (CoAP is still admissible at exactly 1900 B – the hard filter excludes it only strictly above that limit – so it is a legitimate reference here) and a +30.7 % throughput gap against the best logged-candidate reference, gRPC-bidi, while run 2 concentrated on gRPC (90.2 % share) and recovered to near-reference performance (p_{50} 2.397 ms, +1.0 % gap; throughput gap +2.9 %). At 8000 B both repeated runs stayed close

to the isolated-baseline envelope despite different gRPC/gRPC-bidi shares at 128000 B the adaptive runs showed moderate p_{50} gap (≈ +8 %) but a p_{99} that was 8–13 % below the isolated HTTP p_{99} suggesting that looking at median latency alone misses part of the picture, and that the isolated baselines should be read as reference points, not as a head-to-head comparison run at the same time **Fig. 4 (a–d)** shows which protocol was chosen at each decision step for these difficult cases, and **Table 1** puts numbers on how far they were from the baselines. Bad outcome of run-1 is left in deliberately, it is proof that the learner can go wrong early on when it has little data, and leaving it out would make the method look more reliable than it is at startup.

Bayesian prior initialization

The Raspberry Pi Zero 2 W 8000 B comparison gives the clearest evidence that prior initialization changes early adaptive behavior, and it links directly to the warm-start identity (Eq. 21) and (Eq. 22). The no-prior pair converged almost entirely to HTTP (82.8 % and 99.0 % share) and produced higher median and tail RTT; with the prior enabled, the selected-arm distribution shifted toward gRPC/gRPC-bidi. Averaged across the two runs in each condition (n = 2 runs per condition, single host and payload, no confidence intervals – these figures are illustrative of the proof-of-concept), the prior increased

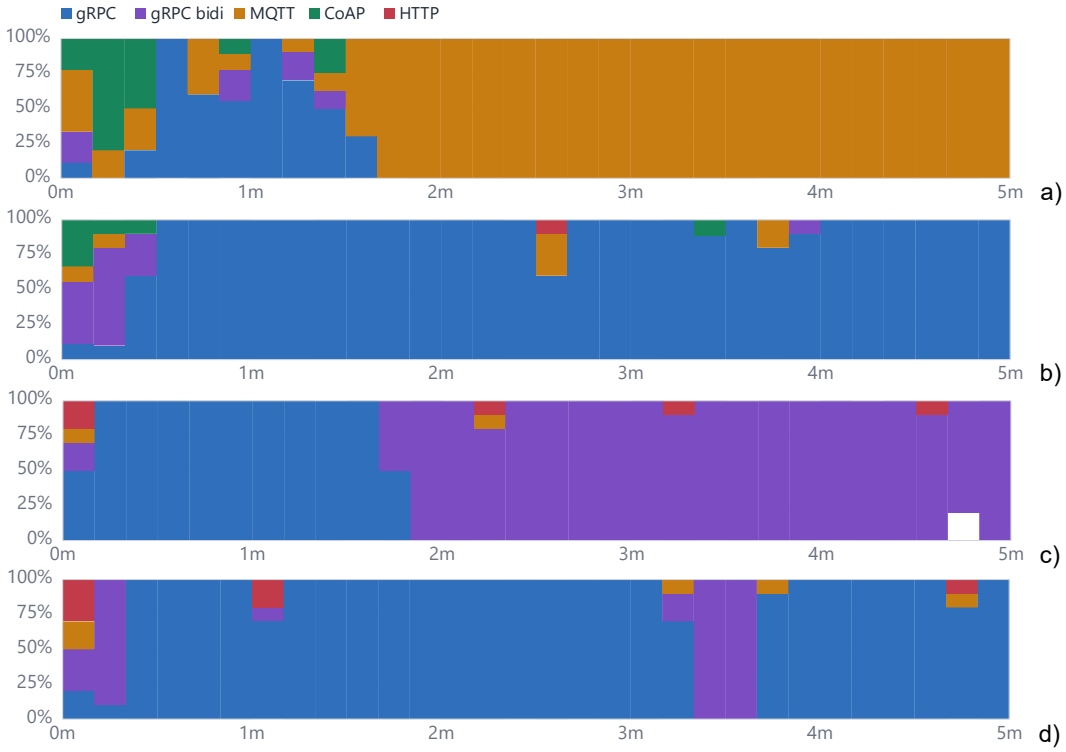

Fig. 4. Selected-arm convergence for the Raspberry Pi 5B hard cases, showing which protocol was chosen at each decision step: (a) 1900 B, cold-start run 1 – selection concentrates on MQTT (≈75.3% share), missing the isolated-baseline envelope (p_{50} 3.271 ms, +37.8% p_{50} gap vs. the CoAP reference); (b) 1900 B, cold-start run 2 – selection concentrates on gRPC (≈90.2% share) and recovers to near-reference performance (p_{50} 2.397 ms, +1.0% gap); (c) 8000 B – selection settles on gRPC-bidi, staying close to the isolated-baseline envelope; (d) 128000 B – selection concentrates on gRPC, with a moderate p_{50} gap (≈+8%) while p_{99} runs 8–13% below the isolated HTTP p_{99}. The (a)/(b) pair shows divergent cold-start trajectories from identical starting conditions.

Table 1. **Selected adaptive runs versus the isolated-protocol baseline reference**

Host	Payload (B)	Run / prior	Dominant arm	p_{50} (ms)	p_{50} gap (%)	Tput gap (%)	p_{99} vs ref. (%)
Pi 5B	1900	run 1, no prior	MQTT 75.3 %	3.271	+37.8	+30.7	+189.5
Pi 5B	1900	run 2, no prior	gRPC 90.2 %	2.397	+1.0	+2.9	+2.2
Pi 5B	8000	run 2, no prior	gRPC-bidi 64.3 %	4.144	+1.4	+1.4	+1.8
Pi 5B	128000	run 2, no prior	gRPC 84.5 %	20.974	+8.3	+5.1	−13.4
Pi Zero 2 W	8000	no prior (mean)	HTTP-dominated	12.401	+14.9..23	+14..20.7	+80..104
Pi Zero 2 W	8000	prior (mean)	gRPC / gRPC-bidi	10.636	−1.3..+5.3	−1.6..+4.6	+27..46

Note: reference = mean of three isolated baseline runs for the same host and payload; positive = worse than reference.

throughput by 19.2 % (70.5 → 84.1 msg/s), decreased p_{50} by 14.2 % (12.40 → 10.64 ms), and decreased p_{99} by 28.9 % (50.43 → 35.87 ms); mean edge CPU rose slightly (8.90 % → 9.26 %), which is reported as an explicit trade-off rather than hidden. **Fig. 5** (per-arm total index) and **Fig. 6** (selected-arm distribution) show that the prior front-loads useful exploitation by initializing each arm from the offline prior, while online evidence still revises the ranking, exactly the graceful-fade behaviour the prior-strength scalar is designed to produce.

Isolated-baseline interpretation and limitations

The results show that no single application protocol is always the best choice. Protocol performance changes with the operating conditions, such as the device, payload size, and measured metric. For this reason, we evaluate the adaptive LinUCB runs against an *isolated baseline envelope*. This envelope is built by taking the best fixed protocol for each host, payload size, and metric from the isolated baseline experiments.

This comparison helps show how close the adaptive selector comes to the best observed fixed protocol result in each case. In many selected runs, the adaptive method performs near this reference envelope. At the same time, the results also show the expected weakness of an online learner at the start of a run: before enough evidence has been collected, early decisions can depend strongly on the first observations.

This baseline should not be interpreted as a direct competitor measured under the same conditions. The fixed-protocol baselines were measured separately and sequentially, while the adaptive selector made decisions inside its own run. Therefore, the baseline envelope is used as a reference for interpretation, not as a true simultaneous counterfactual. For this reason, we do not claim that the adaptive method always outperforms the best fixed protocol.

Five limitations are important in this interpretation.

- *First*, cold-start behavior can lead to poor early choices. The Pi 5B result at 1900 B in run 1 is kept showing that the learning path can depend on the evidence collected at the beginning of the run.

Fig. 5. Per-arm total UCB index over time for the Raspberry Pi Zero 2 W at 8000 B, showing how the Bayesian warm-start reshapes early exploitation: (a) no prior, run 1 – the index trajectories keep HTTP ahead through the cold-start phase, matching the HTTP-dominated selection in Fig. 6-a; (b) no prior, run 2 – the second no-prior run shows the same qualitative HTTP-leading trajectory; (c) with prior, run 1 – the warm-start initializes each arm from the offline prior, front-loading the gRPC/gRPC-bidi indices and shifting the initial policy away from the HTTP-dominated cold trajectory; (d) with prior, run 2 – the second prior-enabled run shows the same qualitative front-loaded gRPC/gRPC-bidi indices, with the warm-start redirecting early exploitation in this run as well.

- *Second*, a Bayesian prior can improve some performance metrics while worsening others. In the Raspberry Pi Zero 2 W 8000 B case, the prior improves throughput, median latency (p_{50}), and tail latency (p_{99}), but slightly increases mean CPU usage. Whether this trade-off is acceptable depends on the system objective. For a CPU-constrained edge device, the operator may prefer a different balance.
- *Third*, the CoAP loss cliff should not be read as a failure of the bandit learner. Instead, it shows that some protocols can become mechanically unsuitable in certain regimes. Such cases must be handled by a hard viability filter that removes unsafe or non-viable actions, rather than relying only on a softer reward penalty.
- *Fourth*, the reported warm-start gains are illustrative rather than statistically characterized. The throughput, p_{50}, and p_{99} improvements above are averaged over only two runs in a single host and payload condition, without confidence intervals or multi-seed replication. A systematic multi-seed study with variance estimates is deferred to the companion paper.

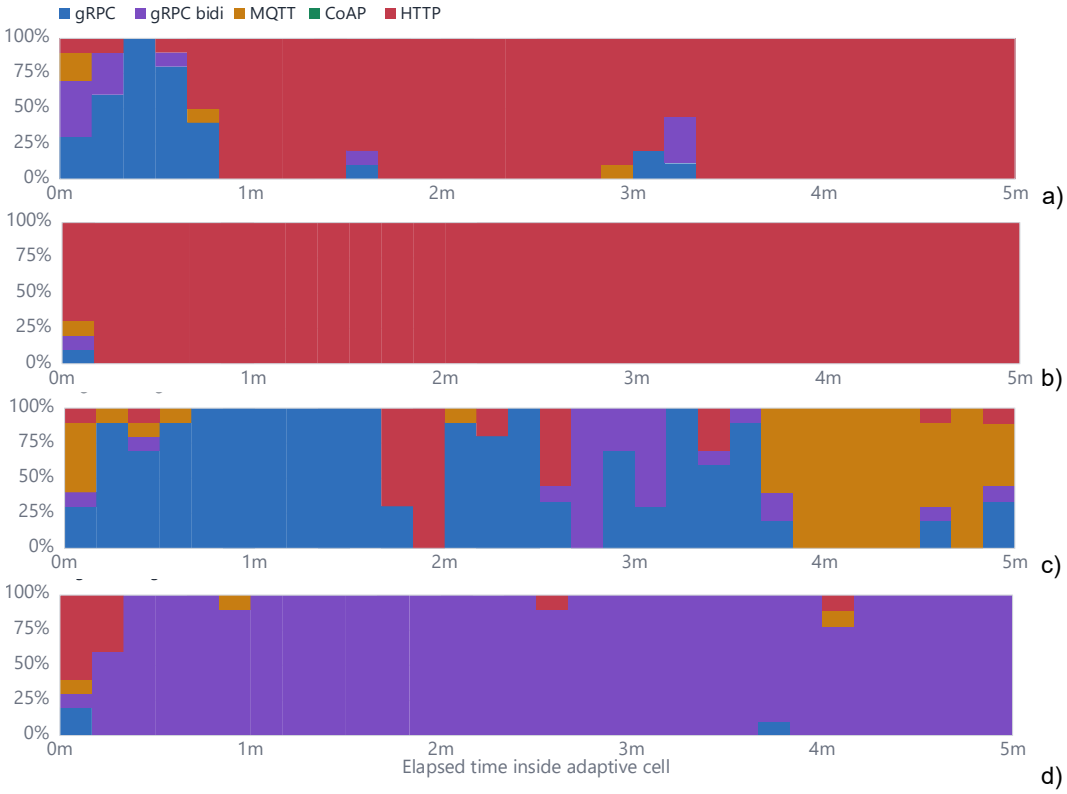

Fig. 6. Selected-arm distribution over time for the Raspberry Pi Zero 2 W at 8000 B, no prior versus Bayesian warm-start: (a) no prior, run 1 – selection collapses almost entirely onto HTTP, reflecting the cold-start trajectory before sufficient online evidence is gathered, with gRPC and gRPC-bidi receiving few selections and yielding higher median and tail RTT; (b) no prior, run 2 – the second no-prior run shows the same qualitative HTTP-dominated pattern; (c) with prior, run 1 – the warm-start shifts the distribution away from HTTP toward gRPC/gRPC-bidi as each arm is initialized from the offline prior, front-loading exploitation while online evidence still revises the ranking, lowering median and tail RTT and raising throughput; (d) with prior, run 2 – the second prior-enabled run shows the same qualitative shift toward gRPC/gRPC-bidi, consistent with the throughput and RTT gains reported across the two-run average.

- *Fifth*, on wired-link deployments (such as tested Raspberry Pi 5B connected via Gigabit Ethernet), the context collapses to [payloadNorm, bias], as noted in the Methods. The worked examples reported here are therefore not an evaluation of the full four-dimensional context under wired conditions: the Pi 5B results reflect only payload-driven arm discrimination, not link-quality discrimination. A wired-aware context extension is planned for the companion paper.

Overall, these examples make the main parts of the approach visible on real hardware: the balance between exploration and exploitation in the UCB score, the effect of Bayesian warm-starting, and the separation between learning and safety constraints. They also define the baseline interpretation used for the broader evaluation in the companion study, including payload and rate sweeps, offline replay experiments, reward-shaping comparisons, and exploration-parameter ablations.

CONCLUSION

We have established the theoretical foundation for an adaptive protocol selection layer for cloud–edge data delivery. Transport choice at adaptive decision ticks is cast as a contextual multi-armed bandit and solved with LinUCB, whose decision index cleanly

separates exploitation (the ridge estimate $\theta^{\mathrm{T}}x$) from exploration (the uncertainty width $\alpha \cdot \sqrt{x^T A^{-1} x}$. The context is a compact four-dimensional vector, the reward captures tail latency, packet loss, and throughput in two bounded formulations an *absolute mode* that scores each protocol against fixed thresholds, and a *relative mode* that scores it against the current field. Non-stationarity is handled by a geometric decay that gradually discounts old observations. A Bayesian warm start lets the selector begin from an offline prior and shift toward live data at a tunable rate. Unsafe protocols are blocked by a hard filter before the learner acts, meaning the safety is not mixed into the reward signal. The implementation supports five concrete transports. Each formula in the paper has a direct counterpart in the code, so the results are fully reproducible given the same configuration and candidate protocols. We validated the formulation on a heterogeneous Raspberry Pi 5B / Raspberry Pi Zero 2 W testbed, showing regime dependence in the baselines, the exploitation/exploration split in the decision logs, and cold-start path dependence in the repeated runs. The companion paper will report the full payload/rate sweep, the replay parameter study, and the reward-shaping and exploration-coefficient sensitivity analyses and will extend the evaluation to dynamic conditions including mid-run wireless network changes and varying payload sizes within a single run. It will also extend the context vector for wired-link deployments – replacing the two Wi-Fi features with wired-link analogues (interface speed, instantaneous goodput, link-saturation indicator) already exposed by the edge reporter. This paper's formulation serves as the fixed methodological baseline for that study.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, authorship, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [R.N., I.B.]; methodology, [R.N.]; software, [R.N.]; formal analysis, [R.N.]; investigation, [R.N.]; writing – original draft, [R.N.]; writing – review and editing, [I.B.]; supervision, [I.B.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Nazarevych, R., & Bolesta, I. (2025). Software of Internet-accessible semiconductor laboratory. *Information Systems and Networks, 17*, 146–159 (In Ukrainian). https://doi.org/10.23939/sisn2025.17.146

[2] gRPC Authors. (2024). *gRPC: A high-performance, open-source universal RPC framework – Documentation*. Cloud Native Computing Foundation. Retrieved May 17, 2026. https://grpc.io/docs/

[3] OASIS. (2019). *MQTT version 5.0* (OASIS Standard). Organization for the Advancement of Structured Information Standards. https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html

[4] Shelby, Z., Hartke, K., & Bormann, C. (2014). *The Constrained Application Protocol (CoAP)* (RFC 7252). Internet Engineering Task Force. https://doi.org/10.17487/RFC7252
[5] Iyengar, J., & Thomson, M. (2021). *QUIC: A UDP-based multiplexed and secure transport* (RFC 9000). Internet Engineering Task Force. https://doi.org/10.17487/RFC9000
[6] Bishop, M. (2022). *HTTP/3* (RFC 9114). Internet Engineering Task Force. https://doi.org/10.17487/RFC9114
[7] Naik, N. (2017). Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. In *2017 IEEE International Systems Engineering Symposium (ISSE)* (pp. 1–7). IEEE. https://doi.org/10.1109/SysEng.2017.8088251
[8] Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning, 47*(2–3), 235–256. https://doi.org/10.1023/A:1013689704352
[9] Lai, T. L., & Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics, 6*(1), 4–22. https://doi.org/10.1016/0196-8858(85)90002-8
[10] Lattimore, T., & Szepesvári, C. (2020). *Bandit algorithms*. Cambridge University Press. https://doi.org/10.1017/9781108571401
[11] Bouneffouf, D., Rish, I., & Aggarwal, C. (2020). Survey on applications of multi-armed and contextual bandits. In *2020 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1–8). IEEE. https://doi.org/10.1109/CEC48606.2020.9185782
[12] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press. http://incompleteideas.net/book/the-book-2nd.html
[13] Slivkins, A. (2019). Introduction to multi-armed bandits. *Foundations and Trends in Machine Learning, 12*(1–2), 1–286. https://doi.org/10.1561/2200000068
[14] Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW '10)* (pp. 661–670). ACM. https://doi.org/10.1145/1772690.1772758
[15] Chu, W., Li, L., Reyzin, L., & Schapire, R. E. (2011). Contextual bandits with linear payoff functions. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)* (Vol. 15, pp. 208–214). PMLR. https://proceedings.mlr.press/v15/chu11a.html
[16] Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55–67. https://doi.org/10.1080/00401706.1970.10488634
[17] Golub, G. H., & Van Loan, C. F. (2013). *Matrix computations* (4th ed.). Johns Hopkins University Press. https://doi.org/10.56021/9781421407944
[18] Abbasi-Yadkori, Y., Pál, D., & Szepesvári, C. (2011). Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems (NeurIPS)* (Vol. 24, pp. 2312–2320). Curran Associates. https://proceedings.neurips.cc/paper/2011/hash/e1d5be1c7f2f456670de3d53c7b54f4a-Abstract.html

[19] Garivier, A., & Moulines, E. (2011). On upper-confidence bound policies for switching bandit problems. In *Algorithmic Learning Theory (ALT 2011), LNCS* (Vol. 6925, pp. 174–188). Springer. https://doi.org/10.1007/978-3-642-24412-4_16

[20] Shivaswamy, P., & Joachims, T. (2012). Multi-armed bandit problems with history. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS)* (Vol. 22, pp. 1046–1054). PMLR. https://proceedings.mlr.press/v22/shivaswamy12.html

[21] Kazerouni, A., Ghavamzadeh, M., Abbasi-Yadkori, Y., & Van Roy, B. (2017). Conservative contextual linear bandits. In *Advances in Neural Information Processing Systems (NeurIPS)* (Vol. 30, pp. 3910–3919). Curran Associates. https://proceedings.neurips.cc/paper/2017/hash/bdc4626aa1d1df8e14d80d345b2a442d-Abstract.html

МЕТОД АДАПТИВНОГО ВИБОРУ ПРОТОКОЛІВ ДЛЯ КОНТЕКСТНО-ЗАЛЕЖНОЇ ПЕРЕДАЧІ ДАНИХ У ХМАРНО-ПЕРИФЕРІЙНИХ СИСТЕМАХ НА ОСНОВІ КОНТЕКСТНОГО БАНДИТА З БАЄСОВОЮ АПРІОРНОЮ ІНІЦІАЛІЗАЦІЄЮ

Роман Назаревич, Іван Болеста
Кафедра радіофізики та комп'ютерних технологій,
Львівський національний університет імені Івана Франка,
вул. ген. Тарнавського, 107, Львів, 79017, Україна

АНОТАЦІЯ

Вступ. Продуктивність протоколів у хмарно-периферійних системах залежить від розміру корисного навантаження, інтенсивності передавання, типу каналу та якості сигналу. Тому протокол, оптимальний за затримкою, втратами або витратами процесорного часу, змінюється разом з умовами роботи, а ранжування за хвостовою затримкою може не збігатися з ранжуванням за медіанною. Статичні механізми й фіксовані евристики не забезпечують належної адаптації, а більшість наявних «адаптивних» рішень є детермінованими оцінювачами, а не системами онлайн-навчання. У статті обґрунтовано адаптивний метод, що формулює вибір протоколу як задачу онлайн-прийняття рішень за часткового зворотного зв'язку. Метод продемонстровано на концептуальних прикладах; повну експериментальну оцінку буде наведено в супровідній статті.

Матеріали та методи. Вибір протоколу сформульовано як задачу контекстного багаторукого бандита для п'яти транспортів і розв'язано алгоритмом LinUCB. Кожен протокол моделюється окремою гребеневою регресією з верхньою довірчою межею, параметри якої обчислюються методом Ґауса. Контекст охоплює логарифмічно нормалізований розмір корисного навантаження, нормалізовані рівень Wi-Fi-сигналу та швидкість передавання, а також вільний член. Винагорода враховує медіанну й хвостову затримку, корисну пропускну здатність і втрати. Нестаціонарність моделюється геометричним згасанням, а баєсова апріорна ініціалізація задає початкові параметри регресії через матрицю точності. Оскільки згасання застосовується покроково, апріорне знання може послаблюватися ще до першої оцінки. Формули узгоджені з програмною реалізацією.

Результати та обговорення. Запропоновану формалізацію для п’яти транспортних протоколів продемонстровано на ізольованих експериментальних прикладах із використанням одноплатних комп’ютерів Raspberry Pi 5B та Pi Zero 2 W. У журналах адаптивного вибору розглядали п’ять протоколів передавання даних: gRPC, gRPC-bidi, MQTT, CoAP і HTTP. Базові експерименти підтвердили, що оптимальний протокол залежить від розміру корисного навантаження, апаратної платформи та цільової функції. Аналіз журналів показав, що в стабільних режимах LinUCB зосереджує вибір на протоколі, найефективнішому для поточного контексту. Водночас приклад невдалого холодного старту продемонстрував чутливість алгоритму до ранніх спостережень. Баєсова апріорна ініціалізація зменшила цей ефект і поліпшила показники затримки та пропускної здатності на початковому етапі.

Висновки. У статті запропоновано відтворюване формулювання адаптивного вибору протоколу для передавання даних між хмарними та периферійними вузлами на основі контекстного багаторукого бандита з баєсовою апріорною ініціалізацією. Визначення контексту, функції винагороди та стратегії дослідження забезпечує незалежне відтворення методу й формує основу для дослідження в супровідній статті.

Ключові слова: контекстний бандит, LinUCB, адаптивний вибір протоколу, хмарно-периферійні системи, баєсовий апріорний розподіл, компроміс дослідження–експлуатація.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
26 May, 2026	25 June, 2026	26 June, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 165–180
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.9:004.8:004.2]-047.27-048.34:519.853

SOFTWARE ENVIRONMENT FOR IDENTIFICATION OF NONLINEAR MATHEMATICAL MODELS USING OPTIMIZATION PROCEDURES

Bohdan Melnyk
Department of Information Systems in Management
Ivan Franko National University of Lviv,
18 Svobody Av., Lviv, 79008, Ukraine

Melnyk, B. (2026). Software Environment for Identification of Nonlinear Mathematical Models Using Optimization Procedures. *Electronics and Information Technologies*, 34, 165–180. https://doi.org/10.30970/eli.34.10

ABSTRACT

Background. For an adequate description of complex systems, it is necessary to use significantly nonlinear mathematical models. An effective approach to the identification of such models is the use of optimization procedures based on various optimization methods. For the comprehensive implementation of these procedures, it is necessary to create software that meets modern trends in the field of computer modeling.

Materials and Methods. To identify models, various optimization algorithms are used, which are developed within the framework of various optimization methods, in particular, within the framework of quasi-Newton methods or stochastic methods. The article considers the BFGS, L-BFGS, and L-BFGS-B algorithms, which belong to the first class of methods, and algorithms that implement the stochastic directed cone method, namely: the basic algorithm and its three modifications. A researcher engaged in mathematical modeling selects one of the software-implemented optimization algorithms that best corresponds to the formulated modeling problem. To do this, he must be able to evaluate the qualitative indicators of these algorithms, which are integrated in a single software environment.

Results and Discussion. The software environment for the identification of mathematical models using optimization procedures, in addition to the specific features associated with identification, must comply with modern programming paradigms and their use. In view of this, it is proposed to use the Python programming environment for its creation. The structure of the software subsystem that implements the selection of the type of mathematical model, its structural and parametric identification and evaluation, as well as integration with other subsystems of the created software environment, is considered. General requirements for ensuring its functionality are formulated. The prospects for the application of cloud IT and AI are considered.

Conclusion. The proposed software and architectural solutions allow creating a software environment that will significantly facilitate the process of computer-generated mathematical models of various complex systems. At the same time, it remains open for further modernization and expansion of functionality.

Keywords: mathematical model, identification, optimization algorithms, software, architecture.

INTRODUCTION

A large number of real objects and processes are considered complex abstract systems, which are described using nonlinear mathematical models. The process of

constructing a mathematical model is multi-stage and significantly depends on the nature of the processes being modeled, as well as on the applied modeling approaches [1, 2].

In the case when it is difficult to describe the structure of the internal connections of the system or it is not advisable from the researcher's point of view, a mathematical model of the “black box” type is used [3]. In the multidimensional case, such a model uses as variables input signals $[v_1, v_2, \dots, v_n]^T = \vec{v}$ (external disturbances that affect the behavior of the system), output signals $[y_1, y_2, \dots, y_m]^T = \vec{y}$, which reflect the external reaction of the system to external influences, state variables $[x_1, x_2, \dots, x_r]^T = \vec{x}$ (formal variables that in a certain way reflect the internal state of the system), as well as equation coefficients $[p_1, p_2, \dots, p_s]^T = \vec{p}$, which reflect the relationship between variables (model parameters).

If a dynamic system is considered, then the values of the variables change over time. In the case of fixing the values of these variables at specific points in time, a discrete model is used. The general form of a discrete model of the “black box” type with constant parameters is as follows

$$\vec{y}^{(k)} = \vec{f}\left(\vec{v}^{(k)}, \vec{x}^{(k)}, \vec{p}\right), \tag{1}$$

where $\vec{y}^{(k)}$, $\vec{v}^{(k)}$, $\vec{x}^{(k)}$ are, respectively, the vectors of output and input signals, as well as the vector of state variables, which are defined at the k-th time instant ($k = \overline{1, N}$, N is the number of time instants at which the values of the vector components are defined); $\vec{p}$ is the vector of model parameters; $\vec{f}(\bullet)$ is a certain vector function, the components of which describe the relationships between the variables.

In the process of building a mathematical model, two important stages can be distinguished among others: the stage of structural identification of the model and the stage of parametric identification of the model [4, 5]. At the stage of structural identification, the form of equations in $\vec{f}(\bullet)$ is selected. This choice significantly depends on the specific formulation of the modeling problem. In the general case, these equations are nonlinear. At the stage of parametric identification, the values of the model parameters are determined. Parametric identification procedures can be carried out either sequentially after the completion of all structural identification procedures, or in parallel with individual ones [5].

Currently, a number of methods are used for structural identification of mathematical models: methods of reduction and complication of the model structure (MSR and MSC), group method of data handling (GMDH), genetic algorithms (GA), methods based on the principles of self-organization of multi-agent systems (SO_MAS) [5]. Promising are the methods of structural identification that use an ontological approach to the construction of mathematical models [6].

In this article, we will limit ourselves to considering the stage of parametric identification of models. Therefore, we will assume that the structure of the model is previously determined on the basis of a detailed study of the behavior of the system or a certain hypothesis about the formal relationships between its input and output signals.

Note that the results published in this article are a continuation of the research presented in [7-11].

Different approaches are used for parametric identification of a mathematical model. In the case of a linear discrete dynamic model in the space of state variables, there is an effective analytical algorithm for parametric identification of the Ho-Kalman, and for a bilinear model, the analytical algorithm proposed by Isidori [4] is used. These algorithms are appropriate to apply at the initial stage of parametric identification of the model if in the structure of the generally nonlinear model it is possible to clearly distinguish the linear [7] or bilinear part. Then, the parameters of these parts of the model obtained at this stage are used as an initial approximation for the identification process of the entire nonlinear model.

For parametric identification of a mathematical model with a significantly complex structure, it is necessary to spend significant computational and especially time resources. This reduces the practical value for the researcher of the modeling process. Therefore, an approach that involves the formal division of a single complex model into simpler submodels that are identified in parallel is quite effective [4, 7]. It is obvious that parallelization of computational procedures accelerates the process of parametric identification of the entire model. And the implementation of algorithms for determining the parameters of individual submodels can hypothetically be significantly simplified. Therefore, the identification process of the general model, even with the need to coordinate the parameters of the submodels, will require less computational cost.

A big problem for significantly nonlinear models is the lack of universal analytical algorithms for their parametric identification. The solution to this problem is the use of an optimization approach. It consists in the fact that as a result of solving the optimization problem, such model parameters are found that allow it to adequately reproduce real processes [8]. Usually, the adequacy of a “black box” model is estimated by the deviation of the real output signals $\vec{y}^{*(k)}$ from the simulated output signals $\vec{y}^{(k)}$. The expression that reflects this deviation defines the objective function in the corresponding optimization problem:

$$Q(\vec{p}) = g\left(\left\|\vec{y}^{*(k)} - \vec{y}^{(k)}\right\|\right) \to \min_{\vec{p}} Q, \tag{2}$$

where $Q(\vec{p})$ is the objective function, the arguments of which are the parameters of the model (1); g is a functional of various types, in particular [8], which reflects the deviation of the real output signals $\vec{y}^{*(k)}$ from the output signals $\vec{y}^{(k)}$ of the model (1).

To solve the optimization problem of type (2), various deterministic and stochastic optimization methods can be used. The choice of a specific method depends on many circumstances [9]. Therefore, the researcher should be able to choose from some methods the one that is most successful for the given problem of parametric identification of a mathematical model. Such a choice involves testing different optimization methods on a certain class of problems [10]. For this, it is necessary to integrate as many algorithms as possible that implement these methods and their modifications into a single software environment [11]. In addition, this environment should be an element of the software package that is directly used for both creating and using the created models. That is why the purpose of this article is to design the architecture of this software environment and develop its individual software modules, which implement and integrate various optimization algorithms used in the process of parametric identification of mathematical models of complex objects and processes.

MATERIALS AND METHODS

The defining element of the considered software complex for the identification of mathematical models is a set of modules that implement optimization algorithms. For example, we will consider only a few of them. The completed implementation of the created software complex should cover as many algorithms as possible.

In this article, we focus on three algorithms that are modifications of the deterministic quasi-Newton method and four variations of the stochastic optimization method. It is important that deterministic methods have their advantages and disadvantages compared to stochastic ones [12]. Therefore, in practical mathematical modeling, there is a need to apply both of these approaches to solving various optimization problems.

Quasi-Newton methods

A feature of quasi-Newton optimization methods that distinguishes them from Newtonian methods is that the Hessian matrix is not directly calculated, but is approximated by another matrix [13]. This, in particular, reduces the computational and time costs of the optimization process. Examples of quasi-Newton optimization algorithms are the classical Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm and its modifications, namely, limited-memory BFGS (L-BFGS) and L-BFGS with box constraints (L-BFGS-B).

BFGS algorithm

As is known, the optimization process is iterative. At each of its iterations, the direction and step of moving to the minimum point of the objective function at the next iteration are determined. The direction and step can be determined directly or indirectly through the parameters of searching for the next point on the trajectory of movement to the minimum, which are inherent to a particular optimization method.

The BFGS algorithm is based on the idea that at each i-th iteration, the inverse Hessian matrix is approximated by the approximate matrix B_i, which is obtained based on the calculation of the first partial derivatives of the objective function [14]. As a result, the direction of further searching for the minimum of the objective function is obtained, and the values of its arguments are calculated at the $i+1$-th iteration. Regarding the optimization problem of type (2), the vector of parameters of the mathematical model at the $i+1$-th iteration is calculated by the formula

$$\vec{p}_{i+1} = \vec{p}_i - \gamma \cdot B_i \cdot g_F(\vec{p}_i), \tag{3}$$

where γ is the optimal step length in the chosen search direction; $g_Q(\vec{p}_i)$ is the gradient of the objective function $Q(\vec{p})$.

L-BFGS algorithm

The basic BFGS algorithm assumes that to determine the matrix B_i at the i-th iteration, it is necessary to store the gradients calculated at all previous iterations. This increases the memory consumption of the computing system and, as a result, slows down the optimization process. In the L-BFGS algorithm, this drawback is eliminated by reducing the number of previous iterations, the results of which must be taken into account at the current iteration [15].

L-BFGS-B algorithm

The L-BFGS-B algorithm is a modification of the L-BFGS algorithm. It provides for imposing restrictions on the arguments of the objective function [16], which makes it possible to make the search for the next point on the path to the minimum more directed.

Software implementation of the quasi-Newton algorithm

The BFGS and L-BFGS-B algorithms are implemented in the scipy.optimize library of the Python environment. The programmatic call of these algorithms is carried out through a universal function [17]

```
minimize(fun, x0, args=(), method=None, jac=None, hess=None, hessp=None,
         bounds=None, constraints=(), tol=None, callback=None, options=None),
```

where the value of parameter `method` ‘BFGS’ or ‘L-BFGS-B’ indicates the corresponding algorithm.

Stochastic directed cone methods

In the directed cone method, at each i-th iteration of the main stage of the optimization process, the further search direction on the way to the minimum point of the objective function is chosen arbitrarily within a certain cone. The vertex of this cone lies at the point

$\vec{p}_i$. The segment of space that limits the cone is determined by its deployment angle Ψ_i and height h_i. Several random test points are selected based on the cone. Among them, the one in which the objective function has the smallest value is chosen. It becomes the vertex of the cone, which is built in the next iteration [4, 8, 10].

Basic algorithm of the directed cone method

The basic algorithm of the directed cone method, proposed by Rastrigin, and its initial adaptation, carried out by Bukashkin [4, 8], assume that the angle of deployment and the height of the cone remain constant at all iterations of the optimization process ($\Psi_i = const, h_i = const, i = 1,2, ...$). This makes this process inefficient [9, 10]. Therefore, algorithms have been proposed that provide for the adaptation of the mentioned optimization parameters to the conditions determined by the current behavior of the objective function.

Algorithm with adaptation of the search step length

The height of the cone h_i basically determines the distance between two neighboring (i-th and $i+1$-th) points on the path of finding the minimum of the objective function. Therefore, h_i can be considered as the search step at the i-th iteration. Depending on the local behavior of the objective function, the current search step should be adapted according to the formula [4, 10]

$$h_{i+1} = h_i \cdot \left(\frac{t_i}{t_{ef}}\right)^{\frac{1}{z-1}}, \tag{4}$$

where $t_i = \tilde{d}/d$ (d is the total number of attempts made with the search step length h_i and $\tilde{d}$ is the number of successful attempts when the objective function decreased); t_{ef} ($0 < t_{ef} < 1$) is a parameter that is set to normalize t_i; z is the dimensionality of the objective function.

In general, the search step should be reduced near the minimum point of the objective function (to determine the minimum as accurately as possible), and at a considerable distance from the minimum point, on the contrary, it should be increased (to speed up the optimization process) [11].

Algorithm with simultaneous adaptation of the cone deployment angle and the search step length

In order to make the optimization process at individual stages, where the objective function allows, more targeted, it was proposed to reduce the cone deployment angle. Such angle adaptation is associated with the need for simultaneous adaptation of the search step length. In this case, a decrease in the cone deployment angle should be accompanied by a corresponding increase in the search step length and vice versa [4, 8].

It has been experimentally proven that the best result of the optimization search is achieved when the adaptation at the i-th iteration of the cone deployment angle and the search step length occurs according to the corresponding formulas [4, 10]

$$\Psi_{i+1} = \delta^{\mu \cdot \alpha - 1} \cdot \Psi_i, \tag{5}$$

$$h_{i+1} = h_i \cdot \left(\frac{t_i}{t_{ef}}\right)^{\frac{1}{z-1}} \cdot \delta^{-(\mu \cdot \alpha - 1)}, \tag{6}$$

where $\delta > 1$ is some constant (the larger the value of this constant, the faster the cone deployment angle adapts); α is a coefficient that characterizes the deviation of the cone axis at two consecutive iterations; μ $(1 \leq \mu \leq 2)$ is some coefficient.

Tunneling algorithm

The tunneling algorithm aims to significantly reduce the time of the optimization process. This is achieved by significantly increasing the search step length in a statistically justified direction on certain sections of the path to the minimum point of the objective function [4, 10].

Software implementation of directed cone method algorithms

For the software implementation of the algorithms of the directed cone method, original software modules were developed. Initially, they were implemented in the Delphi software environment [18]. For example, below is a fragment of a Delphi program for calculating a linear dynamic model with constant parameters in the space of state variables:

$$\begin{cases} \vec{x}^{(k+1)} = F \cdot \vec{x}^{(k)} + G \cdot \vec{v}^{(k)} \\ \vec{y}^{(k+1)} = C \cdot \vec{x}^{(k+1)} + D \cdot \vec{v}^{(k+1)}, \end{cases} \tag{7}$$

where F, G, C, D are the matrices of model parameters of the corresponding dimensions.

```
K: TVector;
F, G, C, D: TMatrix;
NewX: TVector;
VectorDeclare(Params.K, 'K', StateDimension);
MatrixDeclare(Params.F, 'F', StateDimension, StateDimension);
MatrixDeclare(Params.G, 'G', StateDimension, InputDimension);
MatrixDeclare(Params.C, 'C', OutputDimension, StateDimension);
atrixDeclare(Params.D, 'D', OutputDimension, InputDimension);
SetLength(Params.NewX, StateDimension);
var
  i, j, k, l: Integer;
  S: Double;
// Calculation of the vector y
for i := 0 to OutputDimension - 1 do
begin
  S := 0;
  for j := 0 to StateDimension - 1 do
    S := S + Params.C[i][j] * Params.x[j];
{
  for j := 0 to InputDimension - 1 do
    S := S + Params.D[i][j] * v[j];
}
  y[i] := S;
end;

// Calculating the new value of the vector x
for i := 0 to StateDimension - 1 do
begin
  S := Params.K[i];
  for j := 0 to StateDimension - 1 do
    S := S + Params.F[i][j] * Params.x[j];
  for j := 0 to InputDimension - 1 do
    S := S + Params.G[i][j] * v[j];
  Params.NewX[i] := S;
end;
```

```
for i := 0 to StateDimension - 1 do
  Params.x[i] := Params.NewX[i];
```

However, later, in order to adapt to modern programming systems, the optimization algorithms of the directed cone method were modernized and implemented in the Python environment. This made it possible to integrate these algorithms into a single software system with other algorithms that are standardly implemented in this environment, for example, with algorithms of quasi-Newton optimization methods. Note that the created Python version of the software allows you to import previously programmatically implemented models in the Delphi environment, in particular the one presented above. This allows you to use already developed model structures in the Python environment.

Integration of optimization algorithms

For practical modeling, the researcher needs to be given the opportunity to choose one of several optimization algorithms. Therefore, these algorithms must be integrated with each other at the software level. The logic of using integrated optimization algorithms in the process of parametric identification of a mathematical model is reproduced in **Fig. 1**. This scheme considers the possibility of choosing among the following optimization algorithms: two quasi-Newton algorithms – BFGS and L-BFGS-B, and three algorithms of the directed cone method (DCM) – with adaptation of the search step length (DCM-S), with simultaneous adaptation of the cone deployment angle and search step length (DCM-A-S), with tunneling (DCM-T). Note that in the final implementation of the software complex, it is necessary to provide for the integration of other optimization algorithms.

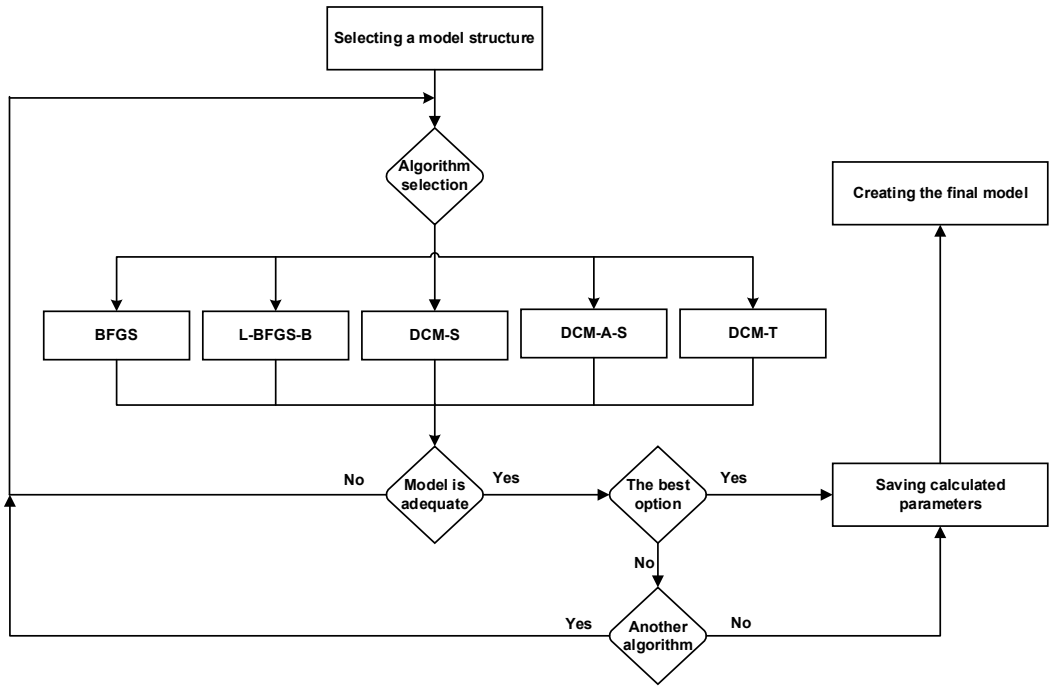

Fig. 1. The logic of using optimization algorithms integrated in a single software package in the process of parametric identification of a mathematical model

After selecting the structure of the mathematical model, the researcher selects one of the listed algorithms. It is used to perform parametric identification of the model. If, according to the criteria selected by the researcher, the model with the calculated parameters does not meet the adequacy conditions, the researcher selects another of the

permissible optimization algorithms. The return to the selection of algorithms can also occur if the researcher considers it necessary to improve the model parameters. In the case when the best option, in the researcher's opinion, is achieved, then the final model is formed.

Note that the proposed integration of optimization algorithms allows us to research the effectiveness of their application on a class of optimization problems using the procedures proposed in [9].

RESULTS AND DISCUSSION

Programming environment requirements

The previously considered approach to the integration of various optimization algorithms should be implemented programmatically as one of the modules of a more complex software system, the purpose of which is to ensure the construction of various nonlinear mathematical models. Other elements of such a system are modules that provide: the system's interconnection with automated complexes for conducting physical experiments, interfaces with information systems, interaction with artificial intelligence (AI), support for databases (DB) and data warehouses (DW), interaction with widespread computer modeling systems, support for parallel computing, carrying out the calculations necessary for the construction and use of mathematical models, and the implementation of a convenient user interface. In addition, some technological requirements are put forward for modern software systems that determine the software environment in which these systems should be created.

In view of this, the programming environment for the developed system of identification of mathematical models should have a number of defining characteristics. In particular [9]: support cross-platform [19] object-oriented [20], generalized [21] and procedural programming [22], the ability to create software in a distributed software environment [23]. For simplicity of software development, it is worth using a high-level programming language [24]. However, in order for the developed software to effectively use the functions of system software, in particular, the functions of the operating system, it is necessary that the programming language has a number of properties of a low-level language [25].

Currently popular programming languages that have the listed properties are Java and C Sharp (C#). The basic constructs of these languages are borrowed from the C++ language [26]. The ISO/IEC 14882:2024 (C++23) standard is valid for C++ [27]. Standardization stimulates leading software manufacturers, in particular such giants of the IT industry as IBM, Oracle, Intel, Microsoft, to create their own compilers from C++, or to support their development within the framework of such projects as LLVM, Embarcadero [28]. The principles on which the concept of the creation and development of this language is based provide the possibility of: implementing any algorithms, using any programming styles, creating an interface with system software, interacting with other programming languages, and creating high-performance software [29]. The latter property is important for reducing time costs in the process of solving optimization problems.

It is also important that the MATLAB software environment, popular among a large number of researchers, is written in C++ and Java [30]. The Simulink visual modeling package integrated into this environment is a convenient tool for simulating the operation of various complex objects described by mathematical models [18].

So, taking into account all the previous considerations, it is appropriate to conclude that it is worth creating the program code of the developed system in a language that is related to C++ or can easily integrate with it.

Despite the significant capabilities of C-oriented programming languages, a large number of software developers, particularly in the field of modeling, prefer Python. The most important reason for the popularity of this language is that it is a high-level language with a simple but at the same time understandable syntax. In addition, it supports cross-platform and various programming concepts, which allows it to be used in different

hardware and software environments to solve different classes of problems [31]. If it is necessary to use low-level capabilities of the C language or previously created C program modules in Python, the C Foreign Function Interface (CFFI) for Python is used [32].

A large number of libraries have been developed for the Python language, which contain various standard functions and fundamental methods for implementing computational procedures. This makes it easier for the programmer to use them when solving specific problems. NumPy [33] and SciPy [34] libraries are used for modeling.

The Python programming environment can be easily integrated with other programming environments used for modeling. For example, in the program code written in Python, you can use the functions and methods of the MATLAB programming and numerical computing platform using the MATLAB Engine API for Python, and vice versa, you can access Python code libraries and constructs from the MATLAB environment by calling a function with the prefix py. [35].

The disadvantage of the Python programming environment is that it works in interpreter mode. This reduces the speed of computational procedures, which negatively affects the effectiveness of the use of optimization methods. To speed up Python programs, the PyPy interpreter is used, the work of which is accelerated by the Just-in-Time compiler [36], or the Codon compiler, created based on the byte-coded LLVM [37] or the similar Numba compiler [38]. In addition to compiling Python code, Codon and Numba provide parallelization of computational procedures, which further speeds up their execution.

Given that the developed software system must interact at the information level with other software systems and software environments, it is necessary to ensure the import and export of data between them. One of the most successful data formats that provides information coordination of different software environments is the csv (comma-separated values) format [39]. The standard of this format [40] also provides support for cross-platform. A file with the .csv extension can be created and edited directly by the user using standard office applications, for example, in the MS Excel spreadsheet environment [39]. This data format must be accepted by the MATLAB environment [41]. Therefore, it should be used to prepare data for model identification and export data to environments that these models use in practice.

Structure of the subsystem for constructing and evaluating a mathematical model

Let us consider the subsystem for constructing and evaluating a mathematical model. It is part of a more general system that performs a wide range of functions – from the processing of experimentally obtained data for modeling to the practical application of the constructed models. **Fig. 2** shows the general structure of the subsystem under consideration.

The subsystem includes the following functional modules:

- Interface module, which provides, on the one hand, user interaction with the subsystem, and on the other hand, the transmission of user control commands to other subsystem modules;
- Repository module is a software-implemented repository of model types defined in the developed software environment or borrowed from outside, which serve as the basis for the final identification of a specific mathematical model. The model type here is understood as the form of the mathematical description of the model (continuous, discrete, differential, integral, interval) and the initial structure of the model (linear or nonlinear with various types of nonlinearity);
- The selection module provides, at the user's request, the selection from the repository of the model type that is most advantageous for a specific modeling task;
- The structural identification module implements a user-defined structural identification algorithm based on the selected model type. In cases where parallel application of structural and parametric identification procedures is envisaged, the structural identification module directly interacts with the parametric identification module;

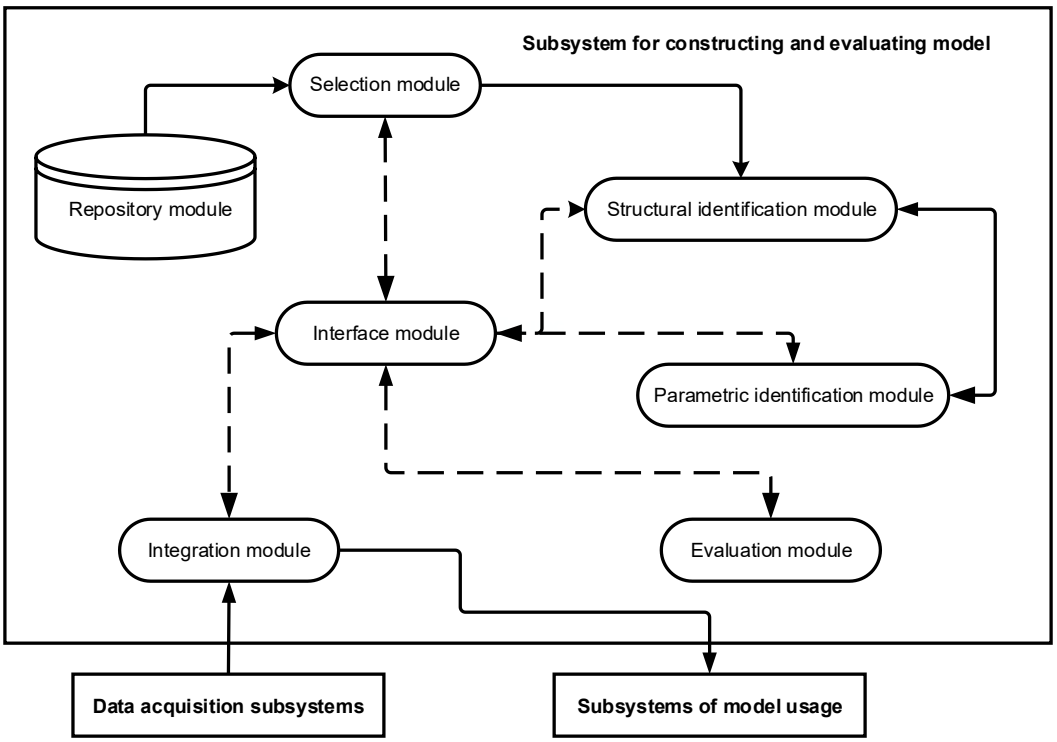

Fig. 2. General structure of the subsystem for constructing and evaluating a mathematical model

- The parametric identification module implements a user-defined parametric identification algorithm. An integral part of this module is the module for implementing optimization algorithms, which was discussed above;
- The evaluation module is used to determine the adequacy and accuracy of the constructed model according to user-defined criteria;
- The integration module provides interaction with other subsystems for:
  - interpretation of experimental data used to build the model (for example, data collected during physical experiments or during computer modeling of the studied processes, in particular in the MATLAB environment);
  - software representation of the constructed model for subsystems that use it in the process of studying the corresponding complex objects.

Technology of using the software environment

The user uses the software environment in interactive mode. The UML use cases diagram is shown in **Fig. 3**.

The software environment allows the user to activate the following main options:

- import data obtained as a result of physical or computational experiments, which are used to identify models;
- export constructed models to other software environments for their practical application;
- filling the repository of model types;
- select the type of models according to the criteria that correspond to the modeling task;
- select the method of structural identification of the model, in particular: MSR [42], MSC [43], GMDH [44], GA [45], SO_MAS [46];

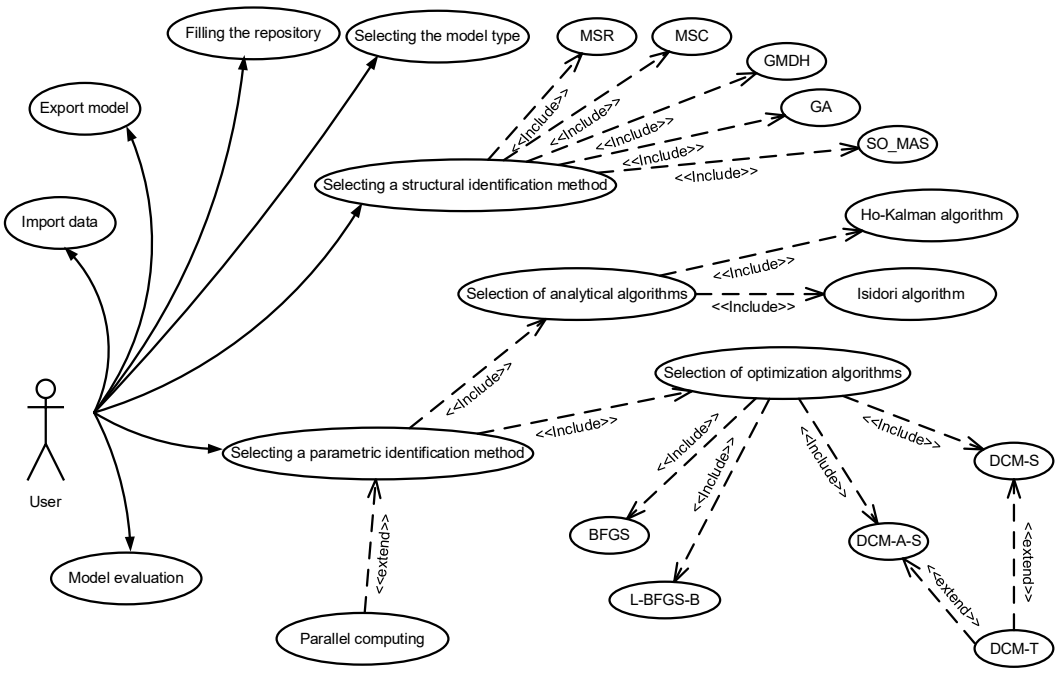

Fig 3. UML Use Case Diagram of Software Operation

- select the method of parametric identification of the model. In the case of using analytical methods, it is possible to select the Ho-Kalman algorithm [47] or the Isidori algorithm [48]. In the case of using optimization methods, the selection of BFGS, L-BFGS-B, DCM-S, and DCM-A-S algorithms is provided. At certain stages of the optimization process, the DCM-S and DCM-A-S algorithms can be supplemented with the DCM-T algorithm. If necessary, during parametric identification of the model, parallel computing algorithms can be used;
- evaluate the constructed model according to certain criteria.

Discussion

Modern software must comply with the latest approaches to its creation and use, as well as IT trends for its future development. Therefore, it is important that the developed software environment is not only functional and user-friendly but also adapted to modern requirements. In particular, it must support cloud technologies (CT) and artificial intelligence (AI) services.

Cloud infrastructure can be implemented on cloud platforms such as Amazon Web Services, Microsoft Azure [49] or Google Cloud Platform [50]. This infrastructure provides services such as scalable cloud computing, cloud data storage, supports MapReduce parallel computing technology [51] and Kubernetes cloud containerization [52], which provides automation of software deployment, scaling and management in a cloud environment, provides RESTful API services [53], which in particular allow you to load data for modeling directly from distributed sensors, and provides integration of the cloud environment with Python libraries, in particular with Matplotlib and Seaborn libraries [54], which allow you to visualize the results of modeling.

Integration of AI tools into the developed software environment increases its intellectualization by adding the function of providing recommendations on modeling procedures, processes of formalized description of models, and their practical use. The integration process is carried out using APIs provided by developers of AI chatbots. For

example, for ChatGPT, OpenAI Corporation provides cloud APIs for various programming environments, in particular for Python [55].

CONCLUSION

This study attempts to propose architectural and software solutions for creating a software environment for identifying mathematical models of complex objects, which is based on the application of an optimization approach. These solutions allow creating software that provides:

- a choice of a wide range of optimization algorithms that implement various optimization methods;
- implementation of all computational stages of mathematical model development, which include structural and parametric identification, parallelization of computational procedures, and evaluation of the resulting model;
- preparation of input data for modeling and creation of software implementation of the model for the environments of its use;
- integration with other software environments and modeling subsystems.

The choice of the Python programming system is justified, as it allows implementing all the designed functionality of the software environment and corresponds to the modern programming paradigm.

Promising directions for future research are formulated, which include the integration of the developed software into the cloud environment, as well as integration with publicly available AI mechanisms.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The author received no financial support for the research, authorship, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The author declares that he has no competing interests.

AUTHOR CONTRIBUTIONS

The author has read and agreed to the published version of the manuscript

REFERENCES

[1] Voronin, A., Lebedeva, I., & Lebedev, S. (2022). A nonlinear mathematical model of dynamics of production and economic objects. *Development Management*, 21(2), 8-15. doi.org/10.57111/devt.20(2).2022.8-15

[2] Batlovskyy, O. O., Foros, G. V., & Siforov, O. I. (2023). *Fundamentals of mathematical modeling.* Odesa: OSUIA (in Ukrainian). https://files.znu.edu.ua/files/Bibliobooks/Inshi78/0058302.pdf (in. Ukr.).

[3] Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., … Hussain, A. (2023). Interpreting Black-Box Models: A Review on Explainable Artificial Intelligence. *Cognitive Computation.* 16, 45–74. doi.org/10.1007/s12-559-023-10179-8

[4] Stakhiv, P. G., Kozak, Yu. Ya., & Hoholyuk, O. P. (2014). *Discrete modeling in electrical engineering and related fields*. Lviv: Publishing House of Lviv Polytechnic (in Ukrainian).

[5] Dyvak, M.P., Pukas, A.V., Porplytsia, N.P., & Melnyk, A.M. (2021). *Applied problems of structural and parametric identification of interval models of complex objects*. Ternopil: University Thought (in Ukrainian). http://dspace.wunu.edu.ua/handle/316497/43614

[6] Dyvak, M.P., Melnyk, A.M., Manzhula, V.I., Spivak, I.Ya., & Porplytsia, N.P. (2024). *Knowledge-oriented systems for identification of interval mathematical models of complex dynamic and static objects*. Ternopil: University Thought (in Ukrainian).
[7] Stakhiv, P., Melnyk, B., Hoholyuk, O., & Trokhaniak, S. (2024). Application of parallel computing technology for modelling complex dynamic objects. *Computational problems of electrical engineering,* 14 (1), 30-35. https://doi.org/10.23939/jcpee2024.01.030
[8] Stakhiv, P., Gogoluk, O., Gamola, O., Melnyk, B., Maday, V., & Melnyk, N. (2023) Application of the Rastrygin's Method in Modeling of Complex Electrical Systems. *2023 24th International Conference on Computational Problems of Electrical Engineering (CPEE)*. Grybów: IEEE. https://doi.org/10.1109/CPEE59623.2023.10285315
[9] Melnyk, B., Kozak, Yu., Melnyk, N., Trokhaniak, S., Dyyak, I., & Yatsyshyn, M. (2025). Research into the Effectiveness of Using the Stochastic Optimization Method in Constructing Discrete Dynamic Models. *2025 15th International Conference on Advanced Computer Information Technologies ACIT'25*. Šibenik: IEEE, 28-34. https://doi.org/10.1109/ACIT65614.2025.11185720
[10] Melnyk, B., Dyvak, M., Melnyk, A., Tkacz, E., Banasik, A., Chwał, J., & Dzik, R. (2025). Application of the Directed Cone Method for the Identification of Mathematical Models of Electromechanical Systems. *Energies*, 18, 5949. doi.org/10.3390/en18225949
[11] Melnyk, B., Stakhiv, P., Melnyk, N., Franko, Yu., Seniv, B., & Martsenyk, Ye. (2024). Software for Implementing the Directed Cone Optimization Method. *2024 14th International Conference on Advanced Computer Information Technologies ACIT'24*. Ceske Budejovice: IEEE, 6-12. https://doi.org/10.1109/ACIT62333.2024.10712522
[12] Fulber-Garcia, V., & Aibin, M. (2024). Deterministic and Stochastic Optimization Methods. *Baeldung*. https://www.baeldung.com/cs/deterministic-stochastic-optimization
[13] Steven, G. J. (2019). Quasi-Newton optimization: Origin of the BFGS update. Notes for 18.335 at MIT. https://github.com/mitmath/18335/blob/spring21/notes/BFGS.pdf
[14] Dai, Y.-H. (2006). Convergence Properties of the BFGS Algoritm. *SIAM Journal on Optimization*. 13 (3). 693-701. https://doi.org/10.1137/S1052623401383455
[15] Singh, M., Shastri, A., & Shaw, K. (2023). *Machine Learning and Optimization for Engineering Design*. Springer Nature. https://doi.org/10.1007/978-981-99-7456-6
[16] Zhu, C., Byrd, R. H., Lu, P., & Nocedal, J. (1997). Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scale Bound-Constrained Optimization. *ACM Transactions on Mathematical Software*, 23 (4), 550–560. https://doi.org/10.1145/279232.279236
[17] scipy.optimize. *SciPy*. https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html#scipy.optimize.minimize
[18] Hoholyuk, O., Kozak, Yu., Rosołowski, E., & Stakhiv, P. (2019). Increasing of the effectiveness of algorithms implementation for development of discrete macromodels and their adaptation to electric circuits simulation programs. *Technical electrodynamics*, 2 (in Ukrainian). https://doi.org/10.15407/techned2019.02.003
[19] Popular programming languages for developing cross-platform applications. *Kotlin* https://kotlinlang.org/docs/multiplatform/programming-languages-cross-platform.html
[20] BasuMallick, Ch. (2022). What is OOP (Object Oriented Programming)? Meaning, Concepts, and Benefits. *Spiceworks*. https://www.spiceworks.com/tech/devops/articles/object-oriented-programming/
[21] Siek, J.G., & Lumsdaine, A. (2011). A language for generic programming in the large, *Science of Computer Programming*. 76 (5), 423-465. https://doi.org/10.1016/j.scico.2008.09.009.

[22] Coursera Staff (2025). Procedural Programming Language: What It Is and When It's Used. *Coursera*. https://www.coursera.org/articles/procedural-programming-language
[23] L'Erario, A., Fabri, J.A., Domingues, A., Duarte, A., Palácios, R., & Pessôa, M. (2012). A Distributed Software Development Environment Dynamics Model. *2012 IEEE Seventh International Conference on Global Software Engineering*, Porto Alegre, 184-184, doi: 10.1109/ICGSE.2012.13.
[24] What is High Level Language? *Geeksfor-Geeks*. https://www.geeksforgeeks.org/software-engineering/what-is-high-level-language/
[25] Şentürk, C. (2024). This Is All You Need to Know About Low-Level Languages. *Tuple*. https://www.tuple.nl/en/blog/all-you-need-to-know-about-low-level-languages
[26] Mensh T. (2026). An In-depth Look at C++ vs. Java. *Toptal*. https://www.toptal.com/developers/c-plus-plus/c-plus-plus-vs-java
[27] ISO. (2024). *ISO/IEC 14882:2024. Programming languages – C++*. https://www.iso.org/standard/83626.html
[28] Rozenberg, Z. (2025). Top C++ compilers in 2025. *Incredibuild*. https://www.incredibuild.com/blog/top-c-compilers
[29] Stroustrup, B. (2020). Thriving in a Crowded and Changing World: C++ 2006-2020. *Proceeding of the ACM Programming Languages*. 4 (HOPL), 7, 1-168 https://doi.org/10.1145/3386320
[30] Schulze, J. (2025). What Is MATLAB? Overview and FAQ. *Coursera*. https://www.coursera.org/articles/what-is-matlab
[31] Haniel, J. (2024)). What is Python used for? 11 most common use cases for Python. *JavaRush*. https://javarush.com/ua/groups/posts/dlja-chogo-vikoristovutjhsja-python
[32] CFFI. *CFFI documentation*. https://cffi.readthedocs.io/en/stable/index.html
[33] NumPy. (2025). *NumPy documentation.* https://numpy.org/doc/stable/
[34] SciPy. (2026). https://scipy.org/
[35] Using MATLAB with Python. *MATLAB*. https://www.mathworks.com/products/matlab/getting-started/using-matlab-python.html#matlab-with-python
[36] PyPy-Features. *PyPy*. https://pypy.org/features.html
[37] Welcome to Codon. *Condon*. https://docs.exaloop.io/
[38] Numba. https://numba.pydata.org/
[39] Shafranovich, Y. (2005). Common Format and MIME Type for Comma-Separated Values (CSV) Files. *SolidMatrix Technologies, Inc*. https://datatracker.ietf.org/doc/html/rfc4180
[40] Cheusheva, S. (2023). How to convert (open or import) CSV file to Excel. *Ablebits.com*. https://www.ablebits.com/office-addins-blog/convert-csv-excel/
[41] Supported File Formats for Import and Export. *MATLAB Help Center*. https://www.mathworks.com/help/matlab/import_export/supported-file-formats-for-import-and-export.html
[42] Obertyuch, R.R., & Slabkyy, A.V. (2025). *Mathematical modeling of mechanical systems*. Vinnytsia: VNTU (in Ukrainian). https://pdf.lib.vntu.edu.ua/books/2025/Obertjukh_2025_119.pdf
[43] Silvestrov, A**.,** Ostroverkhov, M**.,** Spinul, L**.,** Serdyuk, A**.,** & Falchenko, M**.** (2022). Structural and Parametric Identification of Mathematical Models of Control Objects Based on the Principle of Rational Complication. *2022 IEEE 3rd KhPI Week on Advanced Technology (KhPIWeek)*. Kharkiv: IEEE. 1-4. doi.org/10.1109/KhPIWeek57572.2022.9916340
[44] Amiri, M., & Soleimani, S. (2021). ML-based group method of data handling: an improvement on the conventional GMDH. *Complex Intelligent Systems.* 7, 2949–2960. https://link.springer.com/article/10.1007/s40747-021-00480-0#Bib1

[45] McCall, J. (2005). Genetic algorithms for modelling and optimization. *Journal of Computational and Applied Mathematics*, 184 (1), 205-222. https://doi.org/10.1016/j.cam.2004.07.034
[46] Di Marzo Serugendo, G., Gleizes, M.-P., & Karageorgos, A. (2005). Self-organization in multi-agent systems. *The Knowledge Engineering Review*, 20(2), 165-189. doi:10.1017/S0269888905000494
[47] Petreczky, M. (2023). *Realization theory of cyber-physical systems and its application to system identification and model reduction. Optimization and Control [math.OC]*. Université de Lille. https://hal.science/tel-04144797/document
[48] Panos M. Pardalos,P.M., & Yatsenko, V. (2008). *Optimization and Control of Bilinear Systems. Theory, Algorithms, and Applications*. NY: Springer. https://link.springer.com/book/10.1007/978-0-387-73669-3
[49] Perri, D., Simonetti, M., & Gervasi. O. (2022). Deploying Efficiently Modern Applications on Cloud. *Electronics*, 11 (3), 450. https://doi.org/10.3390/electronics11030450
[50] What is Google Cloud Platform (GCP)? *Tandent*. https://ua.talend.com/resources/what-is-google-cloud-platform/
[51] Wang, Y., Goldstone, R., Yu, W., & Wang, T. (2014). Characterization and Optimization of Memory-Resident MapReduce on HPC Systems. *2014 IEEE 28th International Parallel and Distributed Processing Symposium*. Phoenix: IEEE, 799-808, https://doi.org/10.1109/IPDPS.2014.87
[52] Kubernetes And Cloud Native Architecture: A Comprehensive Guide. *Kubegrade*. https://kubegrade.com/kubernetes-cloud-native-architecture/
[53] What is RESTful API? *AWS*. https://aws.amazon.com/what-is/restful-api/
[54] Muddarla, B., & Vatti, P. R. (2024). Machine Learning in Cloud Environments: Leveraging SQL and Python for Big Data Analytics. *Nanotechnology Perceptions*. 20 (7). 12-21. https://doi.org/10.62441/nano-ntp.v20i7.3794
[55] Dyouri, A. (2026). How to Integrate ChatGPT's API With Python Projects. *Real Python*. (19.01.2026). https://realpython.com/chatgpt-api-python/

ПРОГРАМНЕ СЕРЕДОВИЩЕ ДЛЯ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ МАТЕМАТИЧНИХ МОДЕЛЕЙ З ВИКОРИСТАННЯМ ОПТИМІЗАЦІЙНИХ ПРОЦЕДУР

Богдан Мельник
Кафедра інформаційних систем у менеджменті,
Львівський національний університет імені Івана Франка,
Пр. Свободи, 18, Львів, 79008, Україна

АНОТАЦІЯ

Вступ. Для адекватного опису складних систем необхідно використовувати суттєво нелінійні математичні моделі. Ефективним підходом до ідентифікації таких моделей є застосування оптимізаційних процедур, які базуються на різних методах оптимізації. Для комплексної реалізації цих процедур необхідно створити програмне забезпечення, яке відповідає сучасним тенденціям в галузі комп'ютерного моделювання.

Матеріали та методи. Для ідентифікації моделей використовують різні оптимізаційні алгоритми, які розроблені в межах різних методів оптимізації, зокрема у межа квазіньютонівських методів чи стохастичних методів. У статті розглядаються алгоритми BFGS, L-BFGS, L-BFGS-B, що належать до першого класу методів, а також алгоритми, які реалізують метод стохастичного спрямованого конуса, а саме: базовий алгоритм та три його модифікації. Дослідник, який займається математичним моделюванням, вибирає один із програмно реалізованих оптимізаційних алгоритмів, які найкраще відповідають сформульованій задачі моделювання. Для цього він повинен мати змогу оцінити якісні показники цих алгоритмів, які інтегровані у єдиному програмному середовищі.

Результати. Програмне середовище для ідентифікації математичних моделей з використанням оптимізаційних процедур, окрім специфічних особливостей, пов'язаних з ідентифікацією, повинно відповідати сучаснім парадигмам програмування і його використання. Зважаючи на це, запропоновано використати для його створення середовище програмування Python. Розглянуто структуру програмної підсистеми, яка реалізує вибір типу математичної моделі, її структурну і параметричну ідентифікацію та оцінку, а також інтеграцію з іншими підсистемами створюваного програмного середовища. Сформульовані загальні вимоги для забезпечення його функціональності. Розглянуто перспективи застосування хмарних ІТ та ШІ.

Висновки. Запропоновані програмні і архітектурні рішення дають змогу створити програмне середовище, яке суттєво полегшить процес комп'ютерного створення математичних моделей різних складних систем. При цьому воно залишається відкритим для подальшої модернізації і розширення функціональних можливостей.

Ключові слова: математична модель, ідентифікація, оптимізаційні алгоритми, програмне забезпечення, архітектура.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
04 May, 2026	12 June, 2026	15 June, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 181–194
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC 004.89:004.93:004.7

ADAPTIVE SPECTRAL NOISE REDUCTION TO IMPROVE SPEECH RECOGNITION

Oleh Osadchuk**, Ihor Olenych
Department of Radioelectronic and Computer Systems,
Ivan Franko National University of Lviv,
50 Drahomanova Str., Lviv, 79005, Ukraine

Osadchuk, O., Olenych, I. (2026). Adaptive Spectral Noise Reduction to Improve Speech Recognition. *Electronics and Information Technologies*, 34, 181–194. https://doi.org/10.30970/eli.34.11

ABSTRACT

Background. Speech recognition technology is an important component of intelligent systems in various fields, in particular for voice control of smart home devices, personal identification in security systems, automation of various technological processes, etc. Increasing requirements for accuracy, speed, and autonomy of systems require improvement of speech recognition methods, especially when affected by background noise. Therefore, the goal of this work was to develop an adaptive noise reduction method to improve the accuracy of audio information recognition.

Materials and methods. The study of the impact of noise environments on the accuracy of automatic speech recognition covers the stages of spectral analysis of audio signals using short-term Fourier transformation, adaptive noise reduction using soft masks and Wiener filter elements, and signal reconstruction based on inverse Fourier transform. The study uses Whisper speech recognition models and Common Voice and LibriSpeech datasets, supplemented with synthesized noisy recordings.

Results and Discussion. It has been established that various types of background noise (specifically low-frequency stationary, percussive, impulse, tonal, and broadband) and signal-to-noise ratios affect the speech recognition accuracy of Whisper models differently. A method to improve speech recognition accuracy based on adaptive noise reduction is proposed. This method provides flexible suppression of unwanted frequencies without significant speech distortion and allows for the compensation of even complex background noises. A reduction in the average consonant confusion level by 50–60% for the Whisper Tiny model and by 15–30% for the Whisper-1 model in real-time has been demonstrated.

Conclusion. As a result of analyzing the speech recognition efficiency of Whisper models based on phonetic distortion metrics, transcription accuracy, consonant confusion, and artifact perception across a wide range of signal-to-noise ratios, it has been established that the application of adaptive noise reduction provides a significant improvement in accuracy. Despite the higher accuracy of the Whisper-1 model, the Whisper Tiny model with a soft mask demonstrates sufficient efficiency for intelligent systems based on real-time IoT platforms.

Keywords: Whisper models, background noise, noise reduction, Wiener filter, spectral masking, IoT.

INTRODUCTION

The current level of development in electronics, computing, and information technology enables significant progress in modeling natural perception processes, such as hearing, sound signal analysis, and human speech processing. Intelligent sound and voice

command recognition systems hold a vital position in automation, security systems [1], smart homes [2], voice-controlled devices, medical and adaptive technologies [3, 4]. As the scale of these applications grows, the requirements for accuracy, speed, and autonomy of systems are increasing, especially in conditions where processing is performed on low-power IoT platforms in real time [5, 6].

Studies have shown that even modern speech recognition models, such as the Whisper family, demonstrate a significant decrease in accuracy under background noise conditions [7, 8]. This is particularly critical for autonomous security systems, where acoustic interference levels change constantly, and some noises possess complex spectral structures comparable to fricative and affricate consonant sounds. Broadband noise from fans and air conditioners, impulse signals when walking and keyboard clicks, low-frequency vibrations from transport, or the rustling of people moving around in a room overlap critical areas of the frequency spectrum and significantly impair intelligibility (s, sh, ch, zh; “c”, “ш”, “ж”, “ч”) of similar sounds in English and Ukrainian. This leads to recognition errors, which mainly affect consonant groups with similar spectral profiles.

A key feature of this field is that IoT-based security systems mostly operate in real-time on hardware platforms with limited computational resources [9, 10]. Therefore, the use of complex recurrent neural networks or deep noise-reduction models is impractical in such scenarios. Instead, there is a relevant need for spectral processing methods that introduce minimal signal delay (no more than 30–50 ms), require low computational power, and significantly enhance automatic recognition accuracy [11].

Classic spectral subtraction approaches do not provide sufficient quality for practical applications, as harsh truncating of energy in frequency bin cells leads to artifacts known as "musical noise" or "metallic timbre" [12, 13]. These features distort the structure of high-frequency consonants, degrade phase coherence, and can reduce the accuracy of Whisper models, especially lightweight versions like Whisper Tiny [14]. This creates a demand for more flexible frequency masking methods that gradually attenuate unwanted components instead of cutting them off entirely [15].

The aim of this work is an in-depth analysis of the impact of various background noise types on automatic speech recognition accuracy and the development of an adaptive noise reduction method based on the Fast Fourier Transform. Unlike traditional algorithms, the proposed approach uses a gradual transition from hard thresholds to soft spectral masks, similar to Wiener filtering [16, 17]. This reduces noise without losing important spectral characteristics of consonants and minimizes anomalies, preserving the natural sound. Particular attention is focused on noise environments most common in household and office settings, which are critical for real-time speech systems: walking, fan or air conditioner operation, keyboard clicking, indoor rustling, and traffic noise.

The paper analyses which consonant groups in English and Ukrainian are most affected by specific noise types, provides spectral examples, collects confusion statistics for Whisper Tiny and Whisper-1 models, and evaluates the effect of the proposed method. A series of experiments demonstrated that adaptive soft masking based on short-term Fourier transform reduces consonant error rates by 50–60% and significantly improves model accuracy in real-world IoT conditions.

MATERIALS AND METHODS

Physical Meaning and Theoretical Foundation of Spectral Analysis

To study the impact of the noise environment on automatic speech recognition accuracy, a generalized audio signal preprocessing architecture has been developed, covering the stages of spectral analysis, adaptive noise reduction, and signal reconstruction. This approach allows us to investigate the behavior of consonants in

different spectral conditions and determine how energy distortions in the high-frequency regions of the spectrum affect errors in Whisper models of different scales.

Spectral Estimation and Analysis Tools

Since speech is a non-stationary process, the estimation of spectral changes over time was performed using the Short-Time Fourier Transform (STFT). The signal was divided into overlapping frames of length *N* with a hop size *H*, each of which was multiplied by a window function w[*n*]. For a frame with index m (the temporal position of the frame), the spectral representation was calculated as:

$$S_m[k] = \sum_{n=0}^{N-1} x[n + mH]w[n]e^{-j2\pi\frac{kn}{N}}. \quad (1)$$

This representation allowed for tracking energy changes in frequency bins over time. The concept of a "bin" in this context describes a single discrete frequency band f_k, corresponding to the frequency.

$$f_k = \frac{kf_s}{N} \quad (2)$$

This enables the investigation of local frequency regions where fricative and affricate consonants are formed, which are particularly sensitive to noise overlap and determines which specific speech segments are most distorted under the influence of noise

Adaptive Noise Reduction and Masking

The improvement of the noise reduction system was based on the application of adaptive spectral masks. The base mask was constructed similarly to a Wiener filter:

$$G(m,k) = \frac{|S(m,k)|^\alpha}{|S(m,k)|^\alpha + |N(m,k)|^\alpha} \quad (3)$$

Here, $S(m,k)$ is the clean signal spectrum, $N(m,k)$ is the noise spectrum, and a is the sensitivity parameter (*a* = 1 corresponds to the classic Wiener filter, while *a* = 2 is a modified version). The estimation of the noise spectrum $N(m,k)$ was performed in the initial segment of the recording and was adaptively updated thereafter. During frames with low speech activity, the noise spectrum was adjusted using the formula:

$$N_{t+1} = \beta N_t + (1-\beta)|S_t|, \quad (4)$$

The $\beta \in [0.95,0.995]$ value ensured a slow and stable update of the noise model, which is essential for handling fan, air conditioner, and transport noises.

The next crucial component was the time-frequency smoothing of the mask. The mask underwent exponential smoothing in the time domain to reduce sharp transitions between adjacent frames, as well as a moving average in the frequency domain to prevent point-like fluctuations in the spectrum. Additionally, a minimum mask floor of $\approx 0.01 - 0.05$ was introduced to prevent the complete zeroing of narrow frequency bins, thereby reducing the occurrence of "musical" artifacts typical of classic spectral subtraction.

Signal Reconstruction and Synthesis

The inverse transform for each frame is performed using the Inverse Discrete Fourier Transform (IDFT), described by the expression

$$X[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{j2\pi\frac{kn}{N}}, n = 0,1,\dots,N-1 \quad (5)$$

After restoring individual time segments, they are overlaid on each other taking into account the window function, which ensures the correct merging of overlapping segments. The final signal waveform

$$\hat{x}[n] = \frac{\Sigma_m IDFT(S_m)[n-mH]w[n-mH]}{\Sigma_m w^2[n-mH]} \quad (6)$$

This expression guarantees that the signal energy remains constant after summing the overlapping windows. The condition of a non-zero sum of squared windows ensures the absence of artifacts during reconstruction.

Experimental Methodology and Corpus

To conduct experimental studies, a specialized audio corpus was developed based on a combination of high-quality speech standards (Ground Truth) and synthesized noisy samples. The sample was primarily based on recordings from the open datasets Common Voice and LibriSpeech, supplemented by original recordings in a semi-formal style covering security systems, household IoT scenarios, and emergency alerts. Each audio segment, ranging from 3 to 7 seconds in duration, was normalized to a sampling rate of 16 kHz and a 16-bit quantization depth, consistent with the input requirements of the Whisper architecture. Before the additive noise mixing stage, the Root Mean Square (RMS) amplitude of the signals was adjusted to 20 dBFS, providing a consistent baseline for varying the signal-to-noise ratio (SNR) within the range of 0 to 15 dB. The corpus comprised approximately 83% English-language and 17% Ukrainian-language recordings, covering a diverse range of speaker voices across the 850 individual trials. The synthesized noise conditions were produced in two ways: by selecting recordings that already contained real background noise, and by programmatically mixing clean speaker recordings with noise samples sourced from open audio libraries. In **Table 1**, the interval distortion values represent the range observed across SNR levels of 0–15 dB, averaged across all noise instances within each noise category.

The testing methodology involved dividing the corpus into two control groups for a detailed verification of algorithmic robustness. The first group included recordings where target consonants (fricatives, affricates, and plosives) were heavily masked by impulsive, tonal, or low-frequency noise. The second group consisted exclusively of clean reference signals (SNR > 30 dB) passed through an identical processing chain, including spectral subtraction and adaptive soft-mask Wiener filtering. This approach allowed for the verification of the absence of undesirable side effects, such as "musical noise" or spectral degradation, which could trigger new recognition errors in clean speech segments.

In order to minimize biological deviation and take into account the acoustic characteristics of different voices, announcers aged 20 to 55 with different articulation characteristics were involved in the project. The sample included both female voices with a high fundamental frequency (F0≈200-250 Hz), enabling the analysis of noise overlap on high-register harmonics, and male voices (F0≈85-155 Hz) to study robustness against low-frequency transport interference. This demographic and acoustic balance, combined with

850 iterative trials, ensured the formation of a comprehensive factor space for an objective accuracy assessment of the Whisper Tiny and Whisper-1 models under complex operational conditions.

The experiments used Whisper Tiny (39M parameters, multilingual) deployed locally via the faster-whisper library (v1.1.1) for optimized CPU/GPU inference. Cloud-based transcription was additionally evaluated using the Whisper-1 API endpoint accessed via the OpenAI Python SDK. The preprocessing pipeline was implemented in Python 3.10 using numpy (<2.0.0), scipy (1.15.1), torch (2.3.0), and transformers (4.50.1). Audio capture was performed at a sampling rate of 16 kHz with a buffer size of 512 samples (~32 ms per frame). All local inference runs were performed on Raspberry Pi 4. The measured end-to-end processing latency for the adaptive soft-mask Wiener stage on the target IoT hardware was 3-8 ms per audio frame (window length N=512, hop H=128 at 16 kHz), which is within the declared real-time threshold of 30–50 ms.

Analysis of Consonant Distorion Under Noise

The analysis was conducted under various noise conditions (walking, keyboard, fan, transport) with SNR levels of 0, 3, 6, 9, and 15 dB. The results are shown in **Table 1**.

Evaluation Metrics and Experimental Goals

For all experiments, Word Error Rate (WER), Pair Confusion Rate for specific consonant groups, Perceptual Artifact Index, and latency were measured. The set of experiments allowed for the evaluation of each noise reduction stage's effectiveness and the determination of which algorithm characteristics most significantly influence the accuracy of Whisper models in real-world noise environments.

Table 1. **Probability of consonant distortion in various noise environments**

Background Noise	Most Affected Consonants (Eng/Ukr)	Distorted Pair Examples	Tiny Distortion (%)	Whisper-1 Distortion (%)
Footsteps (Impacts)	short percussions, high-frequency noise — tʃ, dʒ, "ч", "дж"	chip vs cheep	3.2–6.0	0.8–2.5
Fan / AC (Tonal + Broadband)	sibilants — s, ʃ, z; "с", "ш", "ж"; high-frequency fricatives	seal vs steel, sheet vs cheat	2.7–5.2	1.8–3.3
Keyboard Clicks (Impulsive)	intermittent back-grounds — p, k, t, tʃ	tap vs cap (in context)	3.4–4.3	2.2–2.6
Passing Traffic (LF & Harmonics)	low-frequency contours — less impact on sibi-lants, more on vowels	morphological errors, insertions	1.2–1.5	0.5–0.9
Indoor Movement / Rustling	sibilants and fricatives	shaft vs sharpen (in context)	0.8–1.7	0.3–1.1

RESULTS AND DISCUSSION

Experimental Setup and Trial Factors

The experimental study comprised 850 individual trials, forming a full factorial space from combinations of two models (Whisper Tiny, Whisper-1) five noise intensity levels, five noise types (walking, fan/AC, keyboard clicks, transport, broadband rustling. The study also

evaluated the impact of a preprocessing stage, specifically utilizing the adaptive soft-mask Wiener method. Each model received identical speech material with controlled consonant groups, enabling the investigation of noise environment impacts on fricatives, affricates, plosives, and their spectral contours. Evaluation was conducted using four key metrics: Phonetic Distortion (Dist), Word Error Rate (WER), Pair Confusion Rate (PCR), and Perceptual Artifact Index (PAI) which made it possible to complexly evaluate the quality of each stage of the noise reduction system.

The results confirmed that different consonant groups react differently to noise conditions and that the nature of the distortions depends on the spectral structure of the noise. Fricatives and affricates consistently showed the greatest distortion under broadband and tonal noise, while plosives (explosive sounds) proved sensitive to impulsive interference, especially at low SNR. The behavior of the Whisper models depended significantly on the spectral characteristics of the noise reduction method. The hard thresholding (flat mask) of spectral subtraction produced high levels of perceptual distortion, whereas the adaptive soft mask with time-frequency smoothing significantly reduced them.

Metrics

Four key metrics were used to evaluate the accuracy of recognition and the quality of audio signal processing, each of which reflects a specific aspect of the interaction between the speech model and the audio stream. The metrics were calculated for each audio fragment, which made it possible to compare the results in a single experimental context and take into account the variability of background noise, duration, and complexity of the material. The signal-to-noise ratio (SNR) served as the experimental condition parameter rather than an evaluation metric per se.

$$SNR_{dB} = 10 \log_{10} \frac{P_{signal}}{P_{noise}} \tag{7}$$

Here, P_{signal} is the average power of the target signal, and $P_{noise}$is the average power of the noise. The SNR value represents the relative intensity of noise compared to the signal. In this study, the SNR was varied from 0 to 15 dB to evaluate the models' performance across different noise conditions.

The Phonetic Distortion percentage was defined as the proportion of incorrectly reproduced phonemes within a specific consonant group:

$$Dist = \frac{N_d}{N_t} \times 100\% \tag{8}$$

Here, N_d is the number of distorted phonemes in the segment, and N_t is the total number of phonemes in that group. This indicator allows us to quantitatively assess the effect of noise on specific phonetic units and the effectiveness of noise reduction algorithms.

Transcription accuracy (Word Error Rate, WER) was calculated according to the standard formula:

$$WER = \frac{S + D + I}{N} \times 100\%. \tag{9}$$

Here, *S* is the number of substitutions, *D* is deletions, *I* is insertions, and *N* is the total number of words in the reference text. WER reflects the overall recognition quality and allows models to be compared under different noise conditions.

The Pair Confusion Rate (PCR) was defined as

$$PCR = \frac{N_{cp}}{N_{ttp}} \times 100\%. \quad (10)$$

Here, N_{cp} is the number of confused phoneme pairs, and N_{ttp} is the total number of tested pairs. This metric evaluates the model's ability to accurately distinguish between similar sounds.

The Perceptual Artifact Index (PAI) evaluates the presence of spectral artifacts following noise reduction, such as resonant "musical noise" and is calculated as a dimensionless coefficient in arbitrary units or in a normalized format from 0 to 1. Lower PAI values correspond to a more natural sound of the processed audio and indicate a smaller amount of undesirable distortion.

Percussive Walking Noise (Short Impacts)

Percussive walking noise functions as high-amplitude, low-frequency, non-stationary impulsive interference, which nevertheless triggers a broadband response within the recording path. Although its primary energy may be low-frequency, the spectral contour of the burst reaching the decoder includes high-frequency harmonics that directly mask the fricative components of the affricates /tʃ/ and /dʒ/.

Table 2. **Effectiveness of the adaptive noise reduction method for short percussions (walking noise)**

Model	SNR, dB	Without noise reduction				With noise reduction			
		Dist, %	WER, %	PCR, %	PAI, a. u.	Dist, %	WER, %	PCR, %	PAI, a. u.
Whisper Tiny	0	6.00	11.2	3.10	0.440	3.90	6.4	1.70	0.352
	3	5.30	10.3	2.90	0.389	2.65	5.7	1.50	0.229
	6	4.60	9.4	2.60	0.356	2.30	5.1	1.30	0.218
	9	3.90	8.3	2.30	0.314	2.54	4.8	1.20	0.233
	15	3.20	6.8	1.90	0.272	2.88	4.1	1.10	0.256
Whisper-1	0	2.50	2.4	1.05	0.230	1.63	1.3	0.48	0.168
	3	2.08	2.1	0.93	0.198	1.04	1.1	0.41	0.125
	6	1.65	1.9	0.82	0.177	0.83	1.0	0.36	0.128
	9	1.23	1.7	0.63	0.153	0.80	0.9	0.31	0.127
	15	0.80	1.4	0.55	0.128	0.72	0.8	0.27	0.120

The Tiny model demonstrates a high "WER no noise reduction (NNR)%" (11.2% at 0 dB, see **Table 2**), indicating its increased error variance in the presence of impulsive noise. This is a consequence of its acoustic space's inability to effectively discriminate the spectral attack of the noise from the acoustic transients of speech.

The use of the Wiener Mask reduces error dispersion, as illustrated by the decrease in PCR with noise reduction (WNR) % from 3.1% to 1.1%. The Wiener Mask utilizes short-term power estimation for each frequency bin. When a walking impulse causes a sudden and localized power surge, the gain coefficient in that specific bin decreases sharply. This prevents the masking of critical transition formants (which indicate the onset or offset of an affricate), allowing the system to restore the correct temporal localization of the phoneme.

Tonal and Broadband Noise (Fan / AC)

Fan noise is a quasi-stationary source that combines tonal components (rotational harmonics) and broadband noise, covering frequencies critical for sibilant fricatives /s/ and

/ʃ/ (primarily 4–8 kHz). This noise causes consistent energetic masking, reducing the local SNR across the entire temporal range of speech.

Table 3. **Effectiveness of the adaptive noise reduction method for tonal and broadband noise (fan)**

Model	SNR, dB	Without noise reduction				With noise reduction			
		Dist, %	WER, %	PCR, %	PAI, a. u.	Dist, %	WER, %	PCR, %	PAI, a. u.
Whisper Tiny	0	5.20	11.3	1.82	0.392	3.380	6.9	0.95	0.288
	3	4.35	10.8	1.52	0.342	2.830	6.1	0.76	0.256
	6	3.95	10.1	1.38	0.318	1.980	5.8	0.63	0.197
	9	3.33	8.9	1.17	0.280	2.160	5.0	0.55	0.148
	15	2.70	7.9	0.97	0.244	2.430	4.6	0.50	0.223
Whisper-1	0	3.30	2.6	0.90	0.278	2.145	1.4	0.45	0.197
	3	2.73	2.4	0.75	0.236	1.774	1.3	0.39	0.163
	6	2.25	2.3	0.63	0.215	1.125	1.2	0.31	0.148
	9	1.98	2.1	0.54	0.197	1.287	1.2	0.29	0.166
	15	1.80	1.9	0.48	0.188	1.620	1.1	0.26	0.168

The "PCR NNR %" in **Table 3** for the Tiny model (1.82% at 0 dB) reflects direct masking, as the noise energy consistently exceeds the fricative discrimination threshold. Since the noise is stationary, the Wiener Mask demonstrates high predictability and effectiveness, achieving steady suppression in constant noise-dominant regions. The reduction in "PCR WNR %" from 0.95% to 0.50% indicates successful spectral separation. The method preserves the internal amplitude modulation of the speech fricative noise above the background, which is key for phoneme identification by the decoder whereas the subtractive method could introduce "musical noise," destroying these vital acoustic features.

Impulsive Noise (Keyboard Clicks)

Keyboard noise is a high-intensity impulsive noise with broad spectral coverage that affects plosive consonants /p/, /k/, /t/, which are characterized by an energy burst and Voice Onset Time (VOT). A click impulse can be erroneously identified by the ASR decoder as a false acoustic transient (i.e., a false burst) or can completely mask the genuine burst, leading to consonant omission or substitution errors.

The high "WER NNR %" (13.1% at 0 dB, see **Table 4**) for the Tiny model indicates that false keyboard transients destabilize the decoder's temporal synchronization. The reduction in "PCR WNR %" from 0.95% to 0.58% results from the successful operation of the Wiener Mask with time-window adaptation. The mask, reacting to the impulse, rapidly reduces its amplitude, minimizing the probability of its false identification as a speech burst. This reduces the energetic correlation between noise and speech in the time domain, restoring the integrity of the VOT for genuine consonants.

Low-Frequency Harmonics (Transport)

Transport noise is stationary and predominantly low-frequency, concentrating on the first and second formants (F1, F2) of vowels. Its primary impact lies in the deformation of vowel frequency contours, which complicates their identification and leads to global decoding errors.

The low "PCR NNR %" (0.45% at 0 dB) in **Table 5** confirms that this noise has a lesser impact on the acoustic accuracy of consonants while the high "WER NNR %" (13.9% at 0 dB) highlights morphological and lexical errors caused by vowel deformation.

The effectiveness of the Wiener Mask against stationary, low-frequency noise is high. The reduction in "WER WNR %" from 11.2% to 4.3% demonstrates the mask's ability to

Table 4. **Effectiveness of the adaptive noise reduction method for impulsive noise (keyboard clicks)**

Model	SNR, dB	Without noise reduction				With noise reduction			
		Dist, %	WER, %	PCR, %	PAI, a. u.	Dist, %	WER, %	PCR, %	PAI, a. u.
Whisper Tiny	0	4.300	13.10	1.46	0.335	2.795	8.40	0.95	0.215
	3	3.950	12.40	1.34	0.323	2.370	6.00	0.73	0.195
	6	3.850	11.80	1.30	0.317	1.925	5.60	0.67	0.155
	9	3.625	10.70	1.26	0.306	2.356	5.40	0.63	0.141
	15	3.400	9.80	1.19	0.292	3.060	5.10	0.58	0.268
Whisper-1	0	2.600	2.60	0.62	0.236	1.690	1.55	0.31	0.140
	3	2.450	2.35	0.57	0.225	1.590	1.30	0.28	0.125
	6	2.400	2.25	0.55	0.221	1.200	1.20	0.26	0.108
	9	2.310	2.10	0.52	0.216	1.500	1.15	0.22	0.125
	15	2.200	1.95	0.50	0.215	1.980	1.05	0.10	0.197

Table 5. **Effectiveness of the adaptive noise reduction method for low-frequency harmonics (transport)**

Model	SNR, dB	Without noise reduction				With noise reduction			
		Dist, %	WER, %	PCR, %	PAI, a. u.	Dist, %	WER, %	PCR, %	PAI, a. u.
Whisper Tiny	0	1.500	13.9	0.45	0.170	0.975	11.2	0.32	0.138
	3	1.425	11.3	0.42	0.162	0.712	6.6	0.25	0.123
	6	1.350	9.4	0.40	0.162	0.675	5.4	0.23	0.118
	9	1.275	8.6	0.38	0.155	0.829	4.7	0.19	0.119
	15	1.200	7.8	0.36	0.152	1.080	4.3	0.18	0.168
Whisper-1	0	0.900	7.1	0.24	0.134	0.585	6.3	0.15	0.125
	3	0.825	5.2	0.22	0.129	0.412	3.4	0.14	0.099
	6	0.750	4.4	0.21	0.125	0.375	2.4	0.12	0.095
	9	0.675	3.7	0.19	0.121	0.439	2.2	0.10	0.099
	15	0.560	3.2	0.17	0.116	0.540	1.9	0.09	0.141

restore the clarity of formant peaks by suppressing constant energy noise in low-frequency bins. This improves the internal acoustic model of vowels within the decoder, significantly increasing the accuracy of word searches in the lexicon and minimizing global errors.

Broadband Non-Stationary Noise (Rustling)

Rustling (indoor movement) is the most challenging type of noise, as it is both non-stationary and broadband.

Its spectral characteristics change rapidly, resulting in the worst "WER NNR %" (17.8% at 0 dB) for the Tiny model, as detailed in **Table 6**. This high error rate reflects the model's inability to effectively process rapidly shifting spectral contours that mask fricatives and other consonants. The effectiveness of the Wiener mask here is noticeable but not as significant as for stationary noise; its performance is markedly lower (with "WER WNR %" decreasing only to 15.9% at 0 dB). This is explained by the fact that noise estimation in an

Table 6. **Effectiveness of the adaptive noise reduction method for broadband non-stationary noise (rustling)**

Model	SNR, dB	Without noise reduction				With noise reduction			
		Dist, %	WER, %	PCR, %	PAI, a. u.	Dist, %	WER, %	PCR, %	PAI, a. u.
Whisper Tiny	0	1.700	17.8	0.38	0.182	1.105	15.9	0.26	0.146
	3	1.530	14.3	0.36	0.172	0.765	8.8	0.23	0.122
	6	1.250	12.1	0.34	0.158	0.625	6.4	0.22	0.116
	9	1.050	10.7	0.32	0.143	0.682	6.0	0.18	0.126
	15	0.800	9.2	0.29	0.128	0.720	5.5	0.17	0.169
Whisper-1	0	1.100	8.9	0.26	0.146	0.715	8.2	0.16	0.121
	3	0.950	6.2	0.24	0.134	0.570	4.3	0.15	0.105
	6	0.800	5.0	0.22	0.128	0.400	2.6	0.14	0.092
	9	0.675	4.1	0.20	0.121	0.439	2.4	0.12	0.099
	15	0.300	3.5	0.18	0.098	0.270	2.2	0.11	0.094

adaptive system requires a certain accumulation time (noise estimation latency). The rate of change in the spectral characteristics of rustling exceeds the mask's adaptation time constant, leading to inaccurate local SNR estimates. Therefore, signal recovery is partial, although the consistent reduction in Dist and PCR still confirms that the Wiener mask provides an overall reduction in energy pollution across the entire spectrum.

Qualitative and Quantitative Assessment of ASR Performance

In most scenarios, such as transport (low-frequency stationary noise), fan (tonal), walking (percussive), and keyboard (impulsive), the proposed noise reduction method ensures full restoration of phonematic accuracy. A qualitative comparison of these effects across different noise types is presented in **Table 7**. This indicates that when noise spectral profiles are relatively stable or predictable (as with tonal or impulsive interference), the adaptive approach effectively reconstructs critical speech features, specifically Voice Onset Time (VOT) and formant structure.

The adaptive soft-mask Wiener filter demonstrates a particular advantage in complex acoustic environments, showing high robustness even with variability in the spectral profile

Table 7. **Qualitative Illustration of Noise Impact and Noise Reduction Method on ASR Decoding**

Ground Truth	Noise (SNR)	Phonematic Substitution Type	ASR before Denoising	ASR after Adaptive Soft-Mask
A large van is speeding	Transport (3 dB)	Vowels /æ/, /ʌ/ (Low)	A large fun is speeding	A large van is speeding
The file must be closed	Fan (7 dB)	Fricatives /f/, /θ/, /v/ (High)	The thigh must be closed	The file must be closed

She will grasp the concept	Walking (5 dB)	Fricatives /s/, /z/ (Aspiration)	She will grass the concept	She will grasp the concept
The dock is far away	Keyboard (8 dB)	Plosives /d/, /g/ (VOT/Burst)	The gock is far away	The dock is far away
He used a winch to pull	Rustling (4 dB)	Affricates /tʃ/, /ʃ/ (Spectral)	He used a witch to pull	He used a winch to pull

of the interference. A characteristic example is rustling (broadband non-stationary noise): due to its capability for precise dynamic modelling, the soft-mask successfully recovers (e.g., witch). This confirms that the mechanism of soft suppression of non-stationary energy is critically important for preserving the integrity of acoustic speech features and ensuring high recognition accuracy.

Based on the aggregated data (**Table 8**), which reflect the average performance of the Whisper Tiny and Whisper-1 models under mean acoustic noise conditions, a consistent conclusion regarding system robustness is established. It is fundamentally established that the Whisper-1 model demonstrates significantly higher intrinsic robustness, as its average "WER NNR %" at 0 dB SNR (4.24%) is 2.65 times lower than that of the Tiny model (11.22%). This difference indicates that the larger capacity of the acoustic model more effectively forms invariant spectral representations capable of reliably segregating speech from noise.

Further analysis confirms a monotonic decrease in all error metrics (Dist, WER, PCR) as the SNR increases, consistent with the normative behavior of ASR systems. The average "PCR NNR %" for Tiny (0.822% at 0 dB) is twice as high as that of Whisper-1 (0.404%), highlighting the smaller model's critical vulnerability to the masking of high-frequency transients and fricatives. This indicator directly correlates with Tiny's inability to accurately localize energy emissions and VOT consonants in the time domain, amplified by background noise.

Table 8. **General assessment of background noise impact using Whisper Tiny and Whisper-1 models speech recognition metrics and the effectiveness of noise reduction method**

Model	SNR, dB	Without noise reduction				With noise reduction			
		Dist, %	WER, %	PCR, %	PAI, a. u.	Dist, %	WER, %	PCR, %	PAI, a. u.
Whisper Tiny	0	2,540	11,22	0,822	0,215	1,651	8,48	0,496	0,157
	3	2,251	9,76	0,728	0,199	1,335	5,50	0,394	0,139
	6	3,000	8,68	0,684	0,191	1,041	4,64	0,350	0,160
	9	2,636	7,78	0,626	0,176	1,205	4,22	0,310	0,153
	15	2,020	5,38	0,490	0,132	1,242	3,04	0,250	0,183
Whisper-1	0	1,580	4,24	0,404	0,158	1,027	3,49	0,214	0,116
	3	1,391	3,23	0,356	0,144	0,869	2,06	0,192	0,098
	6	1,570	3,17	0,486	0,173	0,786	1,68	0,238	0,114
	9	1,374	2,74	0,416	0,161	0,893	1,57	0,208	0,123
	15	0,972	2,11	0,266	0,123	0,882	1,25	0,112	0,120

Evaluation of noise reduction effectiveness through PAI metrics reveals the greatest reduction in objective errors due to the adaptive soft-mask Wiener filter, particularly in the 6 dB SNR range. This fact confirms the principle that ASR optimization should be directed toward preserving acoustic speech features rather than maximizing subjective noise suppression. The Wiener mask minimizes the introduction of spectral artifacts (such as "musical noise") and ensures the restoration of the time-frequency integrity of the speech signal. At high SNR (15 dB), the effectiveness of noise reduction decays asymptotically, as

further processing of a clean signal has limited improvement potential and may even slightly degrade performance due to subtle spectral distortion effects.

CONCLUSION

The work conducted involved an in-depth study of the impact of typical background noise on the accuracy of automatic speech recognition (ASR) based on Whisper models and an assessment of the effectiveness of the proposed adaptive spectral noise reduction method. The evaluated models, Whisper Tiny and Whisper-1, were assessed using Word Error Rate (WER), Pair Confusion Rate (PCR), and Perceptual Artifact Index (PAI) metrics across a wide range of Signal-to-Noise Ratios (SNR).

It was established that the Whisper-1 model demonstrates systematically higher intrinsic robustness, as evidenced by a significantly lower average WER NNR % at low SNR (4.24% at 0 dB), which is 2.65 times lower than that of the Tiny model (11.22%). Its larger capacity enables more effective speech-noise segregation and maintains a lower PCR error level. In contrast, the Whisper Tiny model showed the highest vulnerability to consonant masking and the greatest error dispersion, limiting its application in high noise environments.

The proposed adaptive spectral noise reduction method based on the Wiener soft-mask provided an accuracy boost, reducing the average PCR WNR level by 50–60% for Whisper Tiny and by 15–30% for Whisper-1. The Soft-Mask method is optimal for ASR because it provides low PAI WNR (less distortion) and preserves the integrity of acoustic speech features. An analysis of the metrics leads to the conclusion that the choice of the optimal model depends on accuracy requirements and hardware resources. Although Whisper-1 provides the highest accuracy, its computational cost is excessive for real-time IoT platforms. Considering the need for minimal latency and limited resources, Whisper Tiny — thanks to a significant boost in accuracy (reducing WER by 50% or more) after applying effective Soft-Mask noise reduction — becomes the optimal choice for IoT-based intelligent security systems.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that they have no competing interests.

AUTHOR CONTRIBUTIONS

Conceptualization, [O.O., I.O.]; methodology, [O.O.]; investigation, [O.O.]; writing – original draft preparation, [O.O.]; writing – review and editing, [O.O., I.O.]; visualization, [O.O.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Smart Secure Homes: A Survey of Smart Home Technologies that Sense, Assess, and Respond to Security Threats / J. Dahmen, D. J. Cook, X. Wang, W. Honglei. *Journal of Reliable Intelligent Environments*. 2017. Vol. 3, no. 2. P. 83–98. https://doi.org/10.1007/s40860-017-0035-0

[2] Knowledge-based Cyber Physical Security at Smart Home: A Review / A. Alsufyani, O. Rana, C. Perera. *ACM Computing Surveys*. 2024. Vol. 57, iss. 3. Art. 53. P. 1–36. https://doi.org/10.1145/3698768

[3] Noise Reduction in Industry Based on Virtual Instrumentation / R. Martinek, R. Jaros, J. Baros [et al.]. *Computers, Materials and Continua*. 2021. Vol. 69, iss. 1. P. 1073–1096. https://doi.org/10.32604/cmc.2021.017568
[4] Real-Time Speech Recognition for IoT Purpose using a Delta Recurrent Neural Network Accelerator / C. Gao, S. Braun, I. Kiselev, J. Anumula. *2019 IEEE International Symposium on Circuits and Systems (ISCAS)* : Conference Paper. 2019. https://doi.org/10.1109/ISCAS.2019.8702290
[5] Application of speech recognition technology in IoT smart home / P. Wang, X. Lu, H. Sun, W. Lv. *2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)* : Conference Paper. 2019. https://doi.org/10.1109/IMCEC46724.2019.8984175
[6] Towards Energy-Efficient and Low-Latency Voice-Controlled Smart Homes: A Proposal for Offline Speech Recognition and IoT Integration / P. Huang, I. Ullah, X. Wei, T. A. Ahanger. *arXiv*. 2025. https://doi.org/10.48550/arXiv.2506.07494
[7] Whisper-AT: Noise-Robust Automatic Speech Recognizers are Also *Strong General Audio Event Taggers / Y. Gong, S. Khurana, L. Karlinsky, J. Glass. Proceedings of the 24th Annual Conference of the International Speech Communication Association (INTERSPEECH 2023)*. 2023. P. 2798–2802. https://doi.org/10.21437/Interspeech.2023-2193
[8] Improving Speech Recognition Performance in Noisy Environments by Enhancing Lip Reading Accuracy / D. Li, Y. Gao, C. Zhu [et al.]. *Sensors*. 2023. Vol. 23, iss. 4. Art. 2053. https://doi.org/10.3390/s23042053
[9] AI-Driven Real-Time Distress Detection Through Speech Recognition for Emergency Response Systems / M. Halilaj, E. Myrto, A. F. Dragoni. CEUR Workshop Proceedings (CEUR-WS.org). 2025. Vol. 4044, paper 03. https://ceur-ws.org/Vol-4044/paper03.pdf
[10] IoT based speech recognition system / K. K. Sethy, L. M. Varalakshmi, R. E. [et al.]. *AIP Conference Proceedings*. 2022. Vol. 2393, iss. 1. Art. 020096. https://doi.org/10.1063/5.0074140
[11] Unsupervised Spectral Subtraction for Noise-Robust ASR / G. Lathoud, M. Magimai-Doss, B. Mesot, H. Bourlard. *IDIAP Research Institute*. 2006. https://publications.idiap.ch/downloads/papers/2005/lathoud05e.pdf
[12] Musical Noise Reduction By Processing Spectrogram of Spectral Subtraction Output / H. R. Abutalebi, B. Dashtbozorg. *Proceedings of the 15th International Congress on Sound and Vibration (ICSV15)*. 2008. P. 2979–2986. https://www.researchgate.net/publication/228916452_Signal_Processing_Musical_Noise_Reduction_By_Processing_Spectrogram_of_Spectral_Subtraction_Output
[13] Vaseghi S. V. Advanced Digital Signal Processing and Noise Reduction. 2nd ed. John Wiley & Sons Ltd, 2000. 488 p. ISBN 0-471-62692-9. https://scispace.com/pdf/advanced-digital-signal-processing-and-noise-reduction-375hxslkau.pdf
[14] Wiener Filter and Deep Neural Networks: A Well-Balanced Pair for Speech Enhancement / D. Ribas, A. Miguel, A. Ortega, E. Lleida. *Applied Sciences*. 2022. Vol. 12, iss. 18. Art. 9000. https://doi.org/10.3390/app12189000
[15] Reddy A., Raj B. Soft Mask Methods for Single-Channel Speaker Separation. *IEEE Transactions on Audio, Speech, and Language Processing*. 2007. Vol. 15, iss. 6. P. 1766–1776. https://doi.org/10.1109/TASL.2007.901310
[16] Hout J. van, Alwan A. A novel approach to soft-mask estimation and Log-Spectral enhancement for robust speech recognition. *2012 IEEE International Conference on*

Acoustics, Speech and Signal Processing (ICASSP). 2012. P. 4105–4108. https://doi.org/10.1109/ICASSP.2012.6288821

[17]] Radfar M. M., Dansereau R. M. Single-Channel Speech Separation Using Soft Mask Filtering. *IEEE Transactions on Audio, Speech, and Language Processing*. 2007. Vol. 15, iss. 8. P. 2299–2310. https://doi.org/10.1109/TASL.2007.904233

АДАПТИВНЕ СПЕКТРАЛЬНЕ ШУМОЗНИЖЕННЯ ДЛЯ ПІДВИЩЕННЯ ТОЧНОСТІ РОЗПІЗНАВАННЯ МОВЛЕННЯ

Олег Осадчук*, Ігор Оленич
Кафедра радіоелектронних і комп'ютерних систем,
Львівський національний університет імені Івана Франка,
вул. Драгоманова 50, Львів, 79005, Україна

АНОТАЦІЯ

Вступ. Технологія розпізнавання мовлення є важливою складовою інтелектуальних систем у різних галузях, зокрема, для голосового керування пристроями розумного будинку, ідентифікації особи у системах безпеки, автоматизації різноманітних технологічних процесів тощо. Підвищення вимог до точності, швидкодії та автономності систем потребує удосконалення методів розпізнавання мовлення, особливо за впливу фонових шумів. Тому мета роботи полягала у розробленні адаптивного методу шумозниження для підвищення точності розпізнавання аудіоінформації.

Матеріали та методи. Дослідження впливу шумового середовища на точність автоматичного розпізнавання мовлення охоплює етапи спектрального аналізу аудіосигналів за допомогою короткочасного перетворення Фур'є, адаптивного шумозниження з використанням м'яких масок і елементів Вінер-фільтрації та реконструкції сигналу на основі зворотного перетворення Фур'є. Для дослідження використано моделі розпізнавання мовлення Whisper і набори даних Common Voice та LibriSpeech, доповнені синтезованими зашумленими записами.

Результати. Встановлено, що різні типи фонових шумів (зокрема, низькочастотні стаціонарні, перкусійні, імпульсні, тональні, широкосмугові) і співвідношення сигналу до шуму по різному впливають на точність розпізнавання мовлення моделями Whisper. Запропоновано метод підвищення точності розпізнавання мовлення на основі адаптивного шумозниження, який забезпечує гнучке придушення небажаних частот без суттєвого спотворення мовлення і дає змогу компенсувати вплив навіть складних фонових шумів. Продемонстровано зменшення середнього рівня плутанини приголосних на 50–60% для моделі Whisper Tiny та 15–30% для моделі Whisper-1 у реальному часі.

Висновки. У результаті аналізу ефективності розпізнавання мовлення моделями Whisper за метриками фонетичних спотворень, точності транскрипції, плутанини приголосних та сприйняття артефактів у широкому діапазоні співвідношенням сигналу до шуму встановлено, що застосування адаптивного шумозниження забезпечує суттєве підвищення точності моделей Whisper. Попри вищу точність моделі Whisper-1, модель Whisper Tiny з м'якою маскою демонструє достатню ефективність для інтелектуальних систем на основі IoT-платформ реального часу.

Ключові слова: моделі Whisper, фоновий шум, шумоподавлення, Wiener-фільтр, спектральне маскування, IoT.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
15 April, 2026	09 June, 2026	10 June, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 195–206
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.7

METHODS AND MODELS FOR ENSURING QUALITY OF EXPERIENCE IN INFORMATION AND COMMUNICATION SYSTEMS

Petro Pelekh
Department of Information and Communication Technologies,
Lviv Polytechnic National University,
12 Stepan Bandera Str., Lviv, 79013, Ukraine

Pelekh, P. (2026). Methods and Models for Ensuring Quality of Experience in Information and Communication Systems. *Electronics and Information Technologies*, 34, 195–206. https://doi.org/10.30970/eli.34.12

ABSTRACT

Background. Quality of Experience (QoE) has emerged as a fundamental metric for evaluating the effectiveness of modern information and communication systems, especially in video conferencing environments where real-time interaction is critical. Unlike traditional Quality of Service (QoS) metrics, QoE reflects the subjective perception of users and depends on a combination of technical, behavioral, and contextual factors.

Materials and Methods. The research is based on a comprehensive analysis of contemporary scientific approaches to QoE evaluation in telecommunication systems. A structured set of influencing factors was identified, including technical parameters (network delay, bandwidth, packet loss, noise, and distortions), user-related characteristics (perception, expectations, and interaction behavior), and contextual conditions of service usage. Mathematical modeling methods, including linear regression, neural networks, decision trees, Bayesian approaches, and similarity functions, were considered for predicting QoE. Additionally, the E-model was applied as a standardized approach for evaluating QoS.

Results and Discussion. The experimental results demonstrate that network impairments have a significant negative impact on both QoS and QoE in video conferencing systems. In particular, increased network delays lead to communication interruptions and reduced interactivity, while packet loss results in audio degradation and visual artifacts. The data obtained in this work reveals a strong correlation between objective network parameters and subjective user satisfaction. The conducted modeling confirmed the existence of a nonlinear dependence of QoE on network parameters, particularly delay and packet loss. These results demonstrate the effectiveness of an integrated QoE model in evaluating network services.

Conclusion. The study confirms that network delay and packet loss are among the most influential factors affecting user experience in video conferencing systems. The proposed modeling approach allows for more precise evaluation and forecasting of QoE by incorporating both objective and subjective components. The findings highlight the importance of integrating multiple parameters into unified models for accurate QoE prediction.

Keywords: Quality of Experience, Quality of Service, video conferencing, model, adaptation, communication system.

INTRODUCTION

The QoE (Quality of Experience) model is a tool for assessing the quality of user experience during interaction with technologies and business entities within a specific context, particularly in telecommunication systems. This model takes into account various aspects of the communication ecosystem, such as technical characteristics, business models, user behavior, and usage context.

The QoE model enables the analysis and evaluation of user satisfaction with a particular product or service. It is based on the interaction between the technical parameters of a system and the user's perception of its quality. The main objective of the QoE model is to ensure a high level of user satisfaction and to improve the overall experience of technology use.

This model incorporates various aspects, including audio quality, video quality, data transmission speed, network delay, and other factors. It also considers individual user characteristics, their expectations, and the context in which the technology is used. The QoE model is an important instrument for the design and improvement of telecommunication systems aimed at ensuring a high quality of communication service for end users.

QoE in information and communication technologies plays a key role in ensuring user satisfaction and improving interaction with various services and applications. QoE makes it possible to evaluate Quality of Service (QoS), enhance user interaction, implement traffic management strategies, increase the competitiveness of companies, and improve network monitoring for the rapid detection and resolution of service quality issues before they affect users.

The application of Quality of Experience in information and communication technologies is also important under military conditions. QoE enables the assessment of communication system effectiveness, enhances the productivity and efficiency of military operations, ensures the security and confidentiality of information transmission, and optimizes resource utilization. Such an approach helps to provide reliable communication, fast information exchange, and improved operational efficiency, which are critically important in modern warfare conditions.

MATERIALS AND METHODS

Analysis and Problem Statement

Analyzing the impact of various network parameters on QoE during video conferencing allows the identification of several critical challenges. High network delay leads to interruptions in video transmission and degradation of audio quality, packet loss results in video artifacts and audio dropouts, echo negatively affects speech intelligibility, and network noise introduces distortions that reduce overall sound quality. These factors significantly influence user satisfaction and highlight the need for effective QoE-oriented optimization methods.

In addressing the problem of improving QoE through network parameter optimization, numerous studies have investigated the relationship between technical characteristics and user perception. In particular, the work of Hoßfeld and Pérez [1] proposes a theoretical framework for QoE assessment that integrates both objective measurements and individual user perception. Their approach emphasizes the importance of combining technical monitoring with subjective evaluation to obtain a more accurate representation of user experience.

The study by Seufert et al. [2] focuses on the evaluation and comparison of QoE models using real-world crowdsourced datasets. The authors demonstrate that different

QoE models may produce significantly different results under the same conditions, highlighting the complexity of accurately predicting user experience. Furthermore, Seufert et al. [3] analyze the incompatibility between web and video QoE models, showing that QoE is highly context-dependent and cannot be universally described by a single model.

Schleicher et al. [4] investigate how different stimuli, such as video streaming and web browsing, jointly affect session-level QoE. Their results indicate that QoE cannot be represented as a simple sum of individual components, but rather as a complex interaction of multiple factors. Similarly, Wehner et al. [5] study the impact of content selection strategies on video QoE, particularly in mobile environments, demonstrating that not only network parameters but also content-related factors influence user perception.

In addition, Esper et al. [6] propose QoE-DASH, a tool for evaluating QoE in adaptive streaming systems with edge caching and recommendation mechanisms. Their work highlights the importance of adaptive delivery techniques in maintaining high QoE under varying network conditions. Jelassi and Rubino [7] focus on the development of operative QoE models for IP-based multimedia services, aiming to bridge the gap between theoretical models and practical deployment.

Barakabitze et al. [8] introduce a QoE management architecture for softwarized 5G networks, emphasizing the role of network programmability and dynamic resource allocation in improving user experience. More recent studies, such as d'Arcier et al. [9], explore QoE optimization in distributed XR applications, where real-time performance requirements further increase the sensitivity to network impairments. Similarly, Krogfoss et al. [10] analyze the benefits of 5G and edge computing for QoE in AR/VR environments, demonstrating the importance of low latency and high bandwidth for immersive services.

These studies collectively contribute to the understanding of QoE as a multidimensional concept influenced by technical, contextual, and user-related factors. They highlight the limitations of existing models and the need for more comprehensive approaches that integrate diverse parameters and operating conditions. Based on the analysis of existing scientific works, this study proposes the development of methods and models for optimizing network parameters in order to improve Quality of Experience during video conferencing.

Presentation of the Main Material

The main parameters of the QoE model include various factors that determine the quality of user experience. The key parameters of the QoE model are:

- technical characteristics (data transmission rate, network delay, bandwidth, audio and video quality, noise level, and distortions);
- user behavior (perception of the product or service, level of satisfaction, interaction with the user interface, system response time to user actions);
- usage context (place and time of use, type of device used, specific features of the communication network);
- individual user characteristics (age, gender, experience, etc.);
- interaction model (efficiency and usability of the interface, navigation and information search methods, ease of use of functions and features, level of personalization and individualization of interaction);
- stability and reliability (frequency of errors and failures, recovery time after failures, availability of automatic recovery systems);
- resource consumption (energy usage, computational resource usage, and data volume consumption).

These parameters make it possible to assess Quality of Experience by considering technical, behavioral, and contextual aspects of interaction with technologies and services. By analyzing these parameters, effective strategies can be developed to improve user experience quality and increase user satisfaction with products and services.

Methods for calculating the QoE model may vary depending on the specific usage context and user needs. However, several general approaches can be identified:

User surveys:

- onducting surveys to assess user satisfaction with a product or service;
- using rating scales (e.g., a scale from 1 to 5) to evaluate different aspects of user experience quality.

Objective technical parameters:

- measuring technical characteristics such as data transmission speed, network delay, audio and video quality;
- using specialized software tools to monitor and analyze technical performance indicators.

User behavior analysis:

- tracking user actions within a product or on a website;
- analyzing the time spent performing various tasks and user reactions to different events.

Interaction modeling:

- using mathematical models to predict the impact of various parameters on Quality of Experience;
- applying machine learning methods to analyze large datasets and identify patterns.

Testing with human expert support:

- conducting special tests involving users to evaluate different aspects of experience quality;
- using focus groups and expert assessments to obtain additional feedback.

These methods can be applied individually or in combination to obtain a comprehensive assessment of Quality of Experience in telecommunication systems.

Therefore, this study proposes implementing a method based on mathematical modeling to predict the influence of various parameters on Quality of Experience. This approach involves constructing a mathematical model that reflects the relationship between input parameters (technical, behavioral, contextual) and the output indicator - QoE.

The most common methods for modeling the influence of parameters on QoE include:

- Linear regression. In this method, a linear function is selected that best describes the relationship between input parameters and QoE. For example:

$$QoE = a_1 \cdot n_1 + a_2 \cdot n_2 + \cdots + a_n \cdot n_n + b, \quad (1)$$

  where n_1, n_2,…, n_n are input parameters; a_1, a_2,…, a_n are regression coefficients; and b is the intercept term.
- Neural networks. This method uses artificial neural networks to model complex nonlinear relationships between parameters and QoE. Neural networks can detect complex interactions that cannot be described by linear models.
- Decision trees. In this approach, decision trees are used to identify the most significant parameters influencing QoE. Decision trees can divide input parameters into groups and determine which of them have the greatest impact on Quality of Experience.

- Bayesian methods. Bayesian statistical models are applied to estimate the probability relationships between parameters and QoE. These methods can account for uncertainty and variability in the relationships between parameters.
- Similarity functions. Similarity functions are used to determine the degree of similarity between input parameters and QoE outcomes. They can identify complex heterogeneity in data and determine which parameters have the greatest influence on QoE.

All these methods enable the creation of models capable of effectively predicting Quality of Experience based on various input parameters. Such models contribute to the improvement of products and services in the field of telecommunications, ensuring a high level of service quality for end users.

RESULTS AND DISCUSSION

Based on the QoE model, an experiment was conducted to evaluate the impact of various parameters on user experience quality in telecommunication systems. One possible experimental scenario is described as follows. The objective of the experiment is to determine how network delays affect Quality of Experience during video conferencing. Considering this objective, each stage of the experiment is described in detail below.

Definition of input parameters

Network delay. This parameter is measured in milliseconds and represents the time required to transmit data from the sender to the receiver through the network. Delay measurements are performed using specialized software tools or built-in network monitoring utilities.

Data transmission rate. This parameter is measured as the amount of data that can be transmitted through the network per unit of time (usually in bits per second).

Video and audio quality. Video and audio quality can be measured using various metrics, such as video resolution, audio compression level, noise level, and other indicators.

Preparation of the experimental environment

To conduct the experiment, a testbed with a network featuring controlled delay conditions was created. Software for video conferencing and for monitoring audio and video quality was installed.

Conducting the experiment

Video conferences were carried out under different levels of network delay. By modifying network parameters such as bandwidth and others, different network conditions were reproduced to study their impact on QoE. During the experiment, data on audio and video quality were collected, along with user feedback regarding their experience. The experiment made it possible to determine how network delay affects Quality of Experience and to provide recommendations for improving QoE in video conferencing systems.

Let us examine in more detail each component of the E-model formulas and the formula for evaluating QoE.

1. The E-model for assessing QoS

The E-model formula is expressed as:

$$R = R_0 - I - A - B - C - D, \quad (2)$$

where:
R_0 — initial sound quality, usually measured on a scale from 0 to 100, where 100 represents ideal quality;
I — impact of network delay on sound quality;
A — impact of packet loss on sound quality;
B — impact of echo on sound quality;
C — impact of noise on sound quality;
D — impact of other factors on sound quality.

This model is widely used for assessing speech quality in telecommunication systems and is based on the aggregation of different impairment factors, including noise, delay, packet loss, and equipment-related distortions [11]

Impact of delay (I)

Network delay affects sound quality: the greater the delay, the lower the quality. The delay impact can be calculated as:

$$I = 0{,}024 \cdot t \cdot (95 - R_0), \quad (3)$$

where t is the network delay in milliseconds.

Impact of packet loss (A)

Packet loss also negatively affects sound quality. The impact of packet loss may be calculated as:

$$A = 0.01 \cdot \sum_{i=1}^{n} (95 - R_0) \cdot p_i, \quad (4)$$

where n is the number of lost packets and p_i is the probability of loss of each packet.

Impact of echo (B)

Echo can also degrade sound quality. The impact of echo may be calculated as:

$$B = -0{,}392 \cdot (95 - R_0) \cdot e, \quad (5)$$

where e is the echo level in decibels.

Impact of noise (C)

Noise affects sound quality as well. The impact of noise can be calculated as:

$$C = 0.1 \cdot (95 - R_0) \cdot (1 - 10^{-0.1 \cdot n}), \quad (6)$$

where n is the noise level in decibels.

In formulas (3)-(6), empirical coefficients are introduced that reflect the results of previous studies and represent a generalization of experimentally established dependencies embedded in the classical E-model. In particular, the value 0.024 in formula (3) is a calibration coefficient obtained through statistical analysis of the impact of delay on

perceived speech quality and is used to normalize the model's sensitivity to increasing latency. The value 95 in the same formula (as well as in subsequent expressions) corresponds to the reference level of the maximum, conventionally "ideal" speech quality on the R-scale, which is traditionally used in the E-model as a baseline for assessing quality degradation. Similarly, the other coefficients in formulas (4)–(6) (namely 0.01, 0.392, and 0.1) are of empirical origin and were derived through approximation of subjective speech quality test results under different types of network impairments, allowing the mathematical model to be aligned with actual human perception. Thus, all the introduced coefficients ensure that the model corresponds to experimentally validated relationships between network parameters and subjective quality of experience.

Impact of other factors (D)

Other factors, such as compression and jitter, may also influence sound quality.

2. Formula for evaluating QoE based on the E-model

$$QoE = f(QoS, UserFactors), \tag{7}$$

where:
QoS — service quality calculated using the E-model;
$UserFactors$ — individual user characteristics such as gender, age, experience, etc.;
f — a function that accounts for the interaction between service quality and individual user factors.

The function f may take the following form:

$$QoE = f(QoS, UserFactors) = w_1 \cdot QoS + w_2 \cdot UserFactors, \tag{8}$$

where w_1 and w_2 are weighting coefficients reflecting the importance of service quality and individual user factors in QoE assessment.

Let us introduce the variable λ, representing the intensity of packet loss in the network. In this case, the formula for packet loss impact A becomes:

$$A = 0{,}01 \cdot \lambda \cdot \sum_{i=1}^{n} (95 - R_0), \tag{9}$$

where:
λ — parameter reflecting packet loss intensity in the network;
N — number of lost packets;
R_0 — initial sound quality.

Thus, the variable λ accounts for packet loss intensity, enabling a more accurate assessment of its influence on Quality of Service (QoS) and, consequently, on Quality of Experience (QoE). During the experiment, λ can be used to simulate different network conditions with varying packet loss levels. For example, a series of experiments may be conducted with different values of λ to determine the influence of various packet loss intensities on user experience quality.

Figures 1–2 present the results of modeling the dependence of Quality of Experience on network parameters according to formulas (7)–(8), where QoE is defined as an

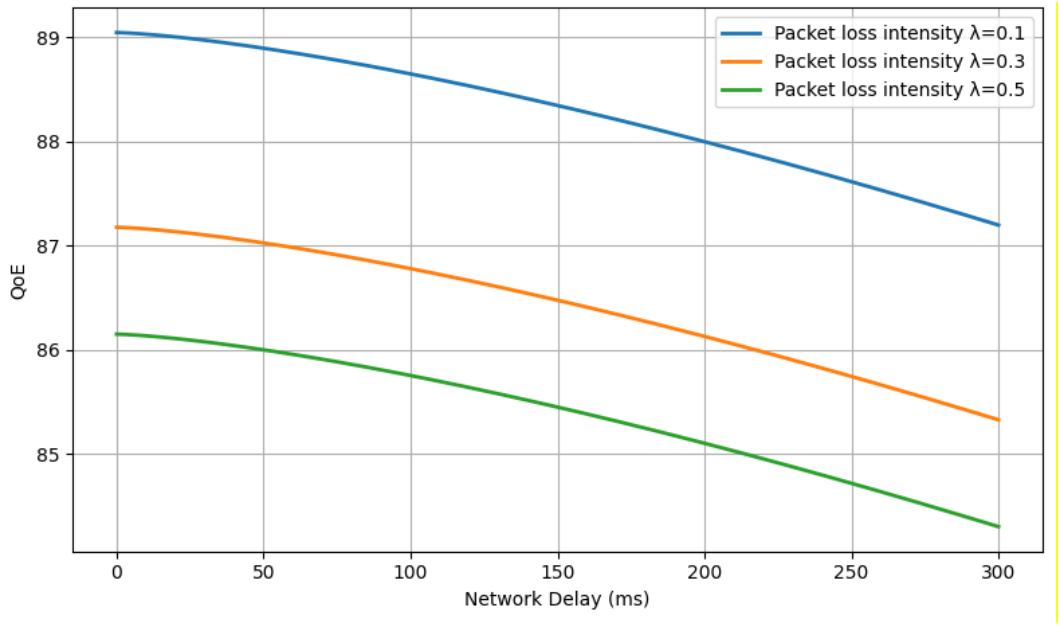

Fig.1. Dependence of QoE on Network Delay

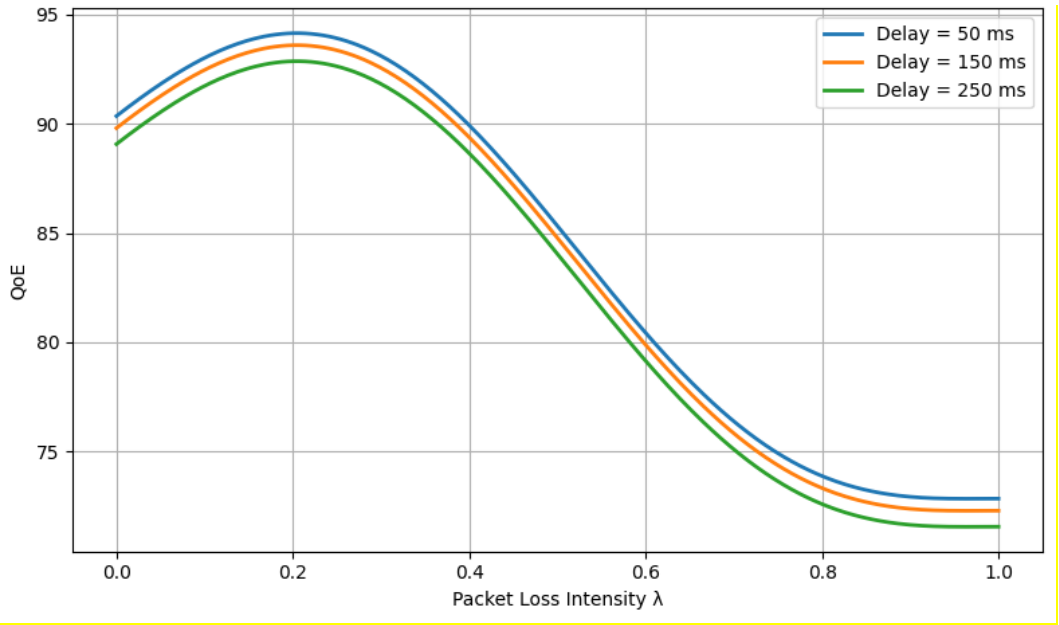

Fig.2. Dependence of QoE on Packet Loss

integrated function of QoS and user-related factors ($UserFactors$). The first graph demonstrates a nonlinear decrease in QoE with increasing network delay under different values of packet loss intensity λ. The obtained results show that even a moderate increase in packet loss leads to a noticeable reduction in QoE at high delay values. The second graph illustrates the dependence of QoE on packet loss intensity under different network delay conditions. It was established that as λ increases, the quality of user experience deteriorates nonlinearly, while the most critical degradation occurs after reaching a certain

threshold value of packet loss. In addition, increasing network delay shifts the QoE curves toward lower values, indicating the combined negative influence of delay and packet loss on perceived service quality.

After conducting the experiment to study the impact of network delay on Quality of Experience during video conferencing, the following results were obtained:

Using the E-model, it was determined that network delays negatively affect audio quality during video conferencing. As delay levels increased, audio quality decreased.

The experimental results also showed that packet loss negatively affects audio quality during video conferencing. As the number of lost packets increased, audio quality deteriorated.

Using the QoE evaluation formula, it was determined that Quality of Experience during video conferencing significantly decreases under conditions of high network delay and high packet loss. This confirms the importance of network optimization to ensure a high level of user experience.

Summarizing the results of the study, it is appropriate to compare the obtained dependencies of QoE on network parameters with well-established standards and scientific approaches. According to the ITU-T Recommendation G.107 (E-model), which is a fundamental standard for assessing voice quality in IP networks, the overall quality metric (R-factor) depends on delay, packet loss, and other network impairment factors. This confirms the results obtained in this study regarding the negative impact of latency and packet loss on QoE. Similarly, ITU-T G.1080, which addresses QoE assessment in video conferencing systems, also emphasizes the critical role of network delay and packet loss in shaping the user's subjective perception of quality. Additionally, WebRTC-based experimental studies [12] demonstrate a strong correlation between objective QoS parameters (delay, jitter, packet loss) and subjective QoE ratings provided by users, which is further supported by experimental research using open datasets for real-time video communication. Taken together, these sources confirm the validity of the obtained results and demonstrate that the proposed approach is consistent with international standards and modern QoE modeling methodologies. In future research, it is planned to extend the proposed QoE model by integrating machine learning methods to achieve more accurate nonlinear approximation of the relationships between network parameters and users' subjective experience. It is also intended to conduct experiments in real-world 5G/edge network environments to validate the model and adapt it to different video conferencing scenarios, including mobile and military applications.

CONCLUSION

Based on the results of the experiment, the following conclusions can be drawn. Network delays and packet loss have a significant impact on both Quality of Service and Quality of Experience in telecommunication systems. To ensure a high level of user experience during video conferencing, further research should focus on network optimization aimed at reducing delays and packet loss. In addition, additional studies are required to determine the optimal network parameters for achieving maximum Quality of Experience.

After conducting the experiment using mathematical models to predict the influence of various parameters on Quality of Experience, it was established that incorporating multiple aspects (such as technical characteristics, network parameters, and individual user preferences) provides a more comprehensive understanding of QoE in information and communication technologies.

The experimental results indicate that the use of such models is effective and enables more accurate prediction of service quality in these technologies. This represents an

important step toward improving user satisfaction, optimizing network performance, and enhancing the efficiency of information and communication systems, particularly under demanding conditions, such as during military operations. The modeling results demonstrated that QoE strongly depends on network delay and packet loss intensity, while the influence of these parameters is nonlinear in nature. The proposed model makes it possible to consider both technical network characteristics and individual user factors for more accurate service quality assessment.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The author received no financial support for the research, writing, and publication of this article.

CONFLICT OF INTEREST

The author declares that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

The author has read and agreed to the published version of the manuscript.

REFERENCES

[1] Hoßfeld, T., & Pérez, P. (2024). A theoretical framework for provider's QoE assessment using individual and objective QoE monitoring. *Proceedings of the 16th International Conference on Quality of Multimedia Experience (QoMEX)*, 235–241. https://doi.org/10.1109/QoMEX61742.2024.10598265

[2] Seufert, A., Wamser, F., Yarish, D., Macdonald, H., & Hoßfeld, T. (2021). QoE models in the wild: Comparing video QoE models using a crowdsourced data set. *Proceedings of the 13th International Conference on Quality of Multimedia Experience (QoMEX)*, 55–60. https://doi.org/10.1109/QoMEX51781.2021.9465422

[3] Seufert, M., Wehner, N., Wieser, V., Casas, P., & Capdehourat, G. (2020). Mind the (QoE) gap: On the incompatibility of web and video QoE models in the wild. *Proceedings of the 16th International Conference on Network and Service Management (CNSM)*, 1–5. https://doi.org/10.23919/CNSM50824.2020.9269041

[4] Schleicher, J., Wehner, N., Hoßfeld, T., & Seufert, M. (2024). (Not) the sum of its parts: Relating individual video and browsing stimuli to web session QoE. *Proceedings of the 16th International Conference on Quality of Multimedia Experience (QoMEX)*, 104–110. https://doi.org/10.1109/QoMEX61742.2024.10598239

[5] Wehner, N., Mertinat, N., Seufert, M., & Hoßfeld, T. (2020). Studying the impact of the content selection method on the video QoE on mobile devices. *Proceedings of the 12th International Conference on Quality of Multimedia Experience (QoMEX)*, 1–4. https://doi.org/10.1109/QoMEX48832.2020.9123088

[6] Esper, J. P., Monção, A. C. B. L., Rodrigues, K. B. C., Both, C. B., Corrêa, S. L., & Cardoso, K. V. (2022). QoE-DASH: DASH QoE performance evaluation tool for edge-cache and recommendation. *Proceedings of the IEEE International Conference on Communications (ICC)*, 757–762. https://doi.org/10.1109/ICC45855.2022.9839234

[7] Jelassi, S., & Rubino, G. (2021). Toward operative QoE models using virtual multimedia IP-based streaming services. *Proceedings of the 24th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, 134–136. https://doi.org/10.1109/ICIN51074.2021.9385556

[8] Barakabitze, A. A., Liyanage, M., & Hines, A. (2020). QoESoft: QoE management architecture for softwarized 5G networks. *Proceedings of the IEEE International Conference on Communications Workshops (ICC Workshops)*, 1–6. https://doi.org/10.1109/ICCWorkshops49005.2020.9145229
[9] d'Arcier, E. F., Jelassi, S., Hirtzlin, P., & Hamza, A. (2026). QoE-based scene adaptation scheme of distributed XR applications. *Proceedings of the IEEE 23rd Consumer Communications & Networking Conference (CCNC)*, 1–2. https://doi.org/10.1109/CCNC65079.2026.11366544
[10] Krogfoss, B., Duran, J., Pérez, P., & Bouwen, J. (2020). Quantifying the value of 5G and edge cloud on QoE for AR/VR. *Proceedings of the 12th International Conference on Quality of Multimedia Experience (QoMEX)*, 1–4. https://doi.org/10.1109/QoMEX48832.2020.9123090
[11] Rodríguez, D. Z., Carrillo, D., Ramírez, M. A., Nardelli, P. H. J., & Möller, S. (2021). Incorporating wireless communication parameters into the E-model algorithm. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. https://doi.org/10.1109/TASLP.2021.3057955
[12] García, B., Gortázar, F., Gallego, M., & Hines, A. (2020). Assessment of QoE for video and audio in WebRTC applications using full-reference models. *Electronics, 9*(3), 462. https://doi.org/10.3390/electronics9030462

МЕТОДИ ТА МОДЕЛІ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ ДОСВІДУ КОРИСТУВАЧА В ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМАХ

Петро Пелех
Національний університет «Львівська політехніка»,
вул. Степана Бандери, 12, Львів, 79013, Україна

АНОТАЦІЯ

Вступ. Якість досвіду користувача (ЯДК, Quality of Experience, QoE) стала фундаментальним показником оцінювання ефективності сучасних інформаційно-комунікаційних систем, особливо в середовищі відеоконференцій, де критично важливою є взаємодія в реальному часі. На відміну від традиційних метрик якості обслуговування (ЯО, Quality of Service, QoS), ЯДК відображає суб'єктивне сприйняття користувачів і залежить від поєднання технічних, поведінкових і контекстуальних чинників.

Матеріали та методи. Дослідження ґрунтується на комплексному аналізі сучасних наукових підходів до оцінювання ЯДК в телекомунікаційних системах. Визначено структурований набір чинників впливу, зокрема технічні параметри (мережева затримка, пропускна здатність, втрата пакетів, шум і спотворення), характеристики користувача (сприйняття, очікування та поведінка під час взаємодії), а також контекстні умови використання сервісу. Для моделювання взаємозв'язку між цими параметрами та ЯДК розглянуто низку математичних і аналітичних методів, зокрема лінійну регресію для виявлення лінійних залежностей, штучні нейронні мережі для моделювання нелінійних зв'язків, дерева рішень для аналізу значущості параметрів, байєсівські моделі для ймовірнісного оцінювання та функції подібності для розпізнавання закономірностей.

Результати. Результати експерименту демонструють, що мережеві порушення мають суттєвий негативний вплив як на ЯО, так і на ЯДК у системах відеоконференцій.

Зокрема, збільшення мережевої затримки призводить до переривань у комунікації та зниження інтерактивності, тоді як втрата пакетів спричиняє деградацію звуку та появу візуальних артефактів. Отримані дані виявляють сильну кореляцію між об'єктивними мережевими параметрами та суб'єктивною задоволеністю користувачів. Проведене моделювання підтвердило наявність нелінійної залежності ЯДК від параметрів мережі, зокрема delay та packet loss. Отримані результати демонструють ефективність використання інтегрованої моделі ЯДК для оцінювання якості мережевих сервісів.

Висновки. Дослідження підтверджує, що мережева затримка та втрата пакетів є одними з найважливіших чинників, які впливають на користувацький досвід у системах відеоконференцій. Запропонований підхід до моделювання забезпечує більш точне оцінювання та прогнозування ЯДК за рахунок урахування як об'єктивних, так і суб'єктивних складових. Отримані результати підкреслюють важливість інтеграції множини параметрів у єдині моделі для підвищення точності прогнозування ЯДК.

Ключові слова: якість досвіду користувача, відеоконференції, модель, адаптація.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
09 February, 2026	11 May, 2026	15 May, 2026	29 June, 2026

Electronics and information technologies, 2026, 34, 207–216
http://publications.lnu.edu.ua/collections/index.php/electronics/index

ELIT

UDC 004.89

DUAL-STREAM MODEL OF LONG SHORT-TERM MEMORY FOR AIR QUALITY PREDICTION USING STATION AND WEATHER DATA

Ivan Rudavskyi* & ***Halyna Klym***
Lviv Polytechnic National University,
12 Stepan Bandera Str., Lviv, 79013, Ukraine

Rudavskyi, I., & Klym, H. (2026). Dual-Stream Model of Long Short-Term Memory for Air Quality Prediction Using Station and Weather Data. *Electronics and Information Technologies*, 34, 207–216. https://doi.org/10.30970/eli.34.13.

ABSTRACT

Background. Air pollution remains a critical environmental and public health issue, requiring accurate and reliable forecasting methods to support decision-making and mitigation strategies. With the increasing availability of environmental data from monitoring stations and meteorological sources, advanced data-driven approaches have become essential for modeling complex air quality dynamics. The development of robust deep learning architectures capable of capturing both temporal dependencies and external environmental influences is of significant importance.

Methods. This study proposes a Dual-Stream Long Short-Term Memory (LSTM) model for air quality forecasting, which processes station-based and meteorological data through two parallel streams. A preprocessing pipeline is applied, including K-nearest neighbors (KNN) imputation for handling missing values and Z-score normalization to standardize input features. The outputs of the LSTM streams are integrated using an attention-based fusion mechanism, which adaptively assigns importance to each data source. The model performance is compared with several baseline approaches.

Results and Discussion. Experimental results demonstrate that the proposed model achieves improved prediction accuracy and stability compared to traditional machine learning methods and single-factor models. The attention-based fusion mechanism effectively captures the dynamic relationships between station data and meteorological conditions, allowing the model to adapt to varying environmental influences. Evaluation using the Index of Agreement, coefficient of determination, and Root Mean Square Error confirms the superiority of the proposed approach in modeling complex air quality patterns.

Conclusion. The proposed Dual-Stream LSTM framework provides an effective solution for integrating heterogeneous environmental data in air quality forecasting tasks. By combining structured preprocessing techniques with an adaptive fusion strategy, the model enhances prediction performance and robustness. The results highlight the potential of multi-source deep learning approaches for environmental monitoring and can be extended to other time-series forecasting applications.

Keywords: air quality prediction; LSTM; dual-stream model; KNN imputation; Z-score; sensor.

INTRODUCTION

Air pollution remains one of the most critical environmental issues worldwide, posing significant threats to human health, ecosystems, and overall quality of life. As urbanization and industrial activities continue to intensify, the need for accurate and timely air quality forecasting becomes increasingly important for environmental monitoring, policy-making, and public health protection. Modern sensing infrastructures, supported by the rapid

development of Internet of Things (IoT) technologies, enable continuous collection of large-scale air quality and environmental data. However, despite the availability of such data, challenges related to data quality, heterogeneity, and complexity still limit the effectiveness of predictive models.

Air quality datasets are inherently influenced by multiple factors, including emission sources, meteorological conditions, and spatial interactions between monitoring stations. In practice, these datasets often suffer from noise, missing values, and inconsistencies caused by sensor malfunctions, communication failures, or environmental disturbances. Such imperfections can degrade model performance and reduce the reliability of forecasting results if not properly addressed during preprocessing and modeling stages.

A wide range of approaches has been explored for air quality prediction, including traditional statistical models [1, 2] and classical machine learning techniques [3, 4]. While these methods can provide reasonable results under certain conditions, they are generally limited in their ability to model complex nonlinear dependencies and long-term temporal patterns present in air pollutant concentrations, such as PM2.5 and PM10. These limitations have motivated the adoption of deep learning methods, which are better suited for capturing intricate relationships in time series data.

Among deep learning approaches, Long Short-Term Memory networks have demonstrated strong performance in sequential prediction tasks due to their capability to retain long-term dependencies and model nonlinear temporal dynamics. This makes LSTM particularly effective for air quality forecasting, where pollutant levels are affected by both historical trends and external influences such as weather conditions. Nevertheless, many existing LSTM-based studies rely on single-source input data or combine multiple factors within a single model without explicitly distinguishing their individual contributions.

To better exploit the multi-source nature of air quality data, it is beneficial to consider separate modeling of different factor groups. In particular, station-based measurements primarily capture temporal dynamics of pollutant concentrations, while meteorological variables (e.g., temperature, humidity, wind speed) represent exogenous factors that significantly influence pollutant dispersion and formation. Treating these inputs independently allows for more specialized feature extraction and reduces the risk of information interference within a single model.

This paper proposes a Dual-Stream LSTM model for air quality forecasting, which processes station data and weather data through two parallel LSTM streams. Each stream is designed to learn distinct patterns associated with its respective input type, and their outputs are subsequently combined to generate the final prediction. This architecture enables more effective modeling of both temporal and exogenous dependencies while preserving the unique characteristics of each data source.

In this work, missing values are handled using a data-driven imputation strategy based on the K-nearest neighbors method, which estimates missing observations by exploiting similarities between data points without relying on strict distributional assumptions. Compared to classical statistical approaches, KNN imputation is more flexible and better suited for complex environmental datasets [5].

In addition, data normalization is applied to ensure stable and efficient model training. Specifically, Z-score normalization is used to standardize input features by removing the mean and scaling to unit variance. This step is particularly important in the proposed framework, as it allows heterogeneous data sources, such as pollutant concentrations and meteorological variables, to be processed in a consistent manner across the dual-stream architecture [6].

The performance of the proposed Dual-Stream LSTM model is evaluated against baseline methods, including traditional machine learning models and standard LSTM architectures. Experimental results demonstrate that the proposed approach more effectively captures complex relationships between air quality and influencing factors, leading to improved forecasting accuracy and robustness.

METHODS

In the course of the process of collecting air quality data, it is possible that monitoring stations may, on occasion, produce incomplete records. Such incompleteness may be due to malfunctions of the sensors, maintenance activities, or communication interruptions. If this issue is not addressed in a satisfactory manner, the accuracy and reliability of predictive models may be adversely affected [7]. To mitigate this issue, the K-nearest neighbors imputation method is used for data imputation.

The K-nearest neighbors imputation method is a non-parametric data reconstruction technique widely used to handle missing values in datasets with complex and unknown distributions. Unlike statistical approaches that rely on predefined assumptions, KNN imputes missing values by identifying the most similar observations in the dataset and using their information to estimate the missing entries. This data-driven strategy allows the method to preserve local structures and relationships within the data, making it particularly suitable for environmental and time-series datasets with nonlinear characteristics. [8]

The core idea of KNN imputation is based on the similarity measurement between data samples. For each instance containing missing values, a set of the k most similar instances (neighbors) is selected according to a distance metric, such as Euclidean distance, computed over the available features. The missing value is then estimated using an aggregation of the corresponding values from these neighbors, typically by calculating their mean or a distance-weighted average. As a result, the imputed values reflect patterns observed in similar data points, ensuring consistency with the underlying data structure.

The KNN imputation process consists of two main stages. In the first stage, distances between samples are computed using only the observed features, and the nearest neighbors are identified for each incomplete instance. In the second stage, the missing values are reconstructed by aggregating the values of the selected neighbors [9]. This approach enables flexible handling of different data types and does not require explicit modeling of the data distribution.

The mathematical formulation of the KNN imputation procedure is provided in **Eq. (1)**, where the missing value is estimated as a function of the values of its nearest neighbors. This formulation allows the method to adapt to local data patterns and improves the overall completeness and reliability of the dataset for subsequent modeling tasks.

$$x_i^m = \sum_{j \in N_k(i)} \left(w_{ij} \cdot x_j^m\right) \Big/ \sum_{j \in N_k(i)} w_{ij}\,, \tag{1}$$

where $w_{ij} = [d(i,j) + \varepsilon]^{-1}$, $d(i,j)$ is the distance between sample i and neighbor j, ε is a small constant to avoid division by zero, x_j^m is the value of feature m in neighbor j.

Normalization Process: Air quality and meteorological variables often exhibit different scales and units, which can adversely affect the convergence and stability of machine learning models. To mitigate this issue, Z-score normalization is applied to standardize each feature by removing the mean and scaling to unit variance. [10] This transformation ensures that all input variables contribute proportionally during model training, preventing features with larger numerical ranges from dominating the learning process. Z-score normalization has been shown to improve model convergence, enhance stability, and increase the accuracy of forecasting predictions. The formula for Z-score normalization is presented in **Eq. (2)**:

$$x_i' = x_i - \mu/\sigma\,, \tag{2}$$

where x_i is the original value of the feature, μ is the mean of the feature over the training data, σ is the standard deviation of the feature, x_i' is the normalized value.

To effectively integrate the information extracted from the two data streams, an attention-based fusion mechanism is employed. Instead of directly combining the outputs of the LSTM models, the proposed approach assigns adaptive importance weights to each stream, allowing the model to dynamically determine the contribution of station-based and meteorological features to the final prediction.

h_s and h_w denote the hidden representations produced by the LSTM models for station data and weather data, respectively. The attention mechanism computes a set of weights that reflect the relative importance of each representation. These weights are then used to construct a fused feature vector, which captures both temporal dependencies and external influences. The attention weights are calculated as presented in **Eq (3)**:

$$\begin{aligned} a_s &= \exp(e_s)/\exp(e_s) + \exp(e_w), \\ a_w &= \exp(e_w)/\exp(e_s) + \exp(e_w), \end{aligned} \tag{3}$$

where e_s and e_w are the attention scores for the station and weather streams, respectively. These scores are typically obtained through a learnable function, such as a fully connected layer:

$$e_s = w^T h_s, \quad e_w = w^T h_w, \tag{4}$$

where h_s and h_w are the hidden feature representations obtained from the LSTM models for station data and meteorological data, respectively, w^T is a learnable weight vector of the attention layer, e_s and e_w are the corresponding attention scores reflecting the importance of each data stream.

The fused feature representation h_f is then computed as a weighted combination of the two feature vectors:

$$h_f = \alpha_s h_s + \alpha_w h_w, \tag{5}$$

where h_s and h_w are the feature vectors from the station and meteorological streams, respectively, α_s and α_w are the normalized attention weights assigned to each stream.

The fused feature vector is passed through a fully connected layer to generate the final prediction output.

This attention-based fusion strategy enables the model to adaptively emphasize either temporal patterns from station data or external meteorological influences depending on the current environmental conditions. Compared to simple concatenation or static weighting methods, the proposed approach provides greater flexibility and improves the model's ability to capture complex relationships in air quality data.

RESULTS AND DISCUSSION

The workflow of the proposed LSTM-based air quality prediction is presented in **Fig. 1**.

Figure 2 shows a comparison of the predicted and measured PM2.5 concentrations, and **Figure 3** shows the corresponding comparison for PM10 concentrations. The results demonstrate the effectiveness of the proposed method in modelling air pollutant behavior, as it produces predictions that closely match the observed values.

The Index of Agreement (IA) is a normalized statistical metric used to quantify the degree of correspondence between predicted and observed values. Its values range from 0 to 1, where 1 indicates perfect agreement between model outputs and observations, while values closer to 0 reflect poor predictive performance [11].

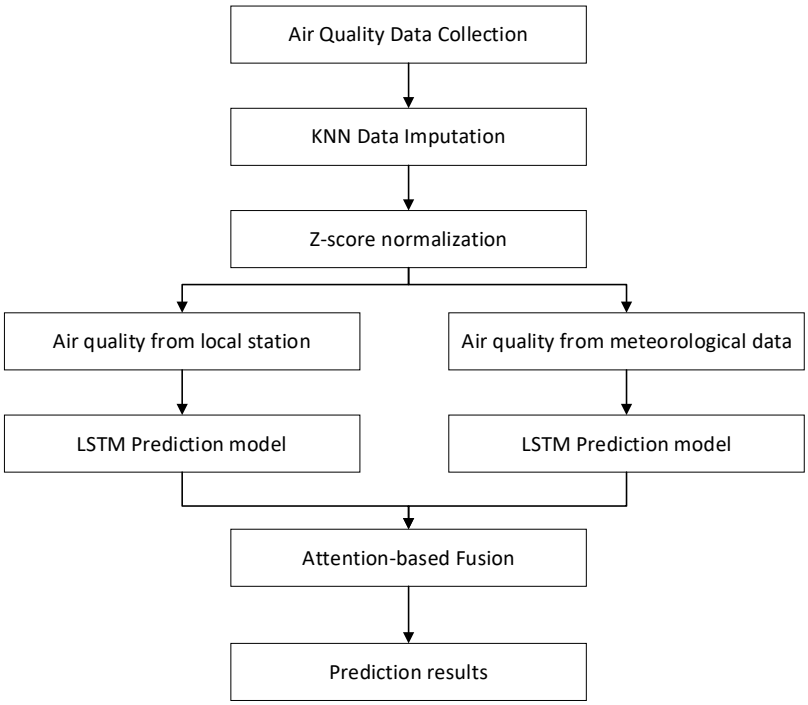

Fig 1. Workflow of the proposed LSTM-based air quality prediction

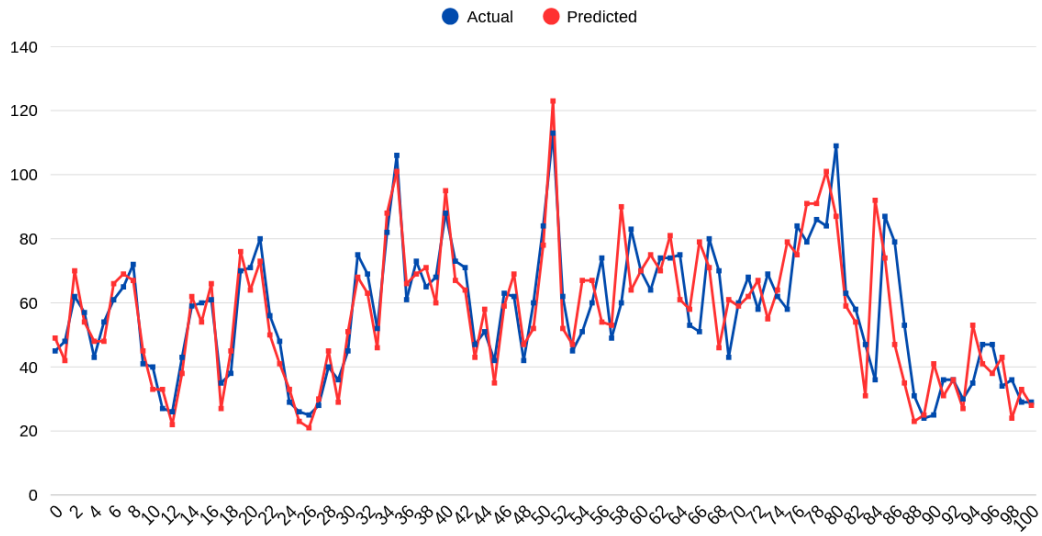

Fig 2. The comparison of predicted and actual values for the PM2.5 parameter

The IA evaluates model accuracy by comparing the magnitude of prediction errors to the total potential deviation in the observed data. Unlike simple error metrics, it accounts for both systematic and random discrepancies, providing a more comprehensive measure of model performance. This makes it particularly suitable for assessing environmental prediction models, where variability and uncertainty are inherent in the data.

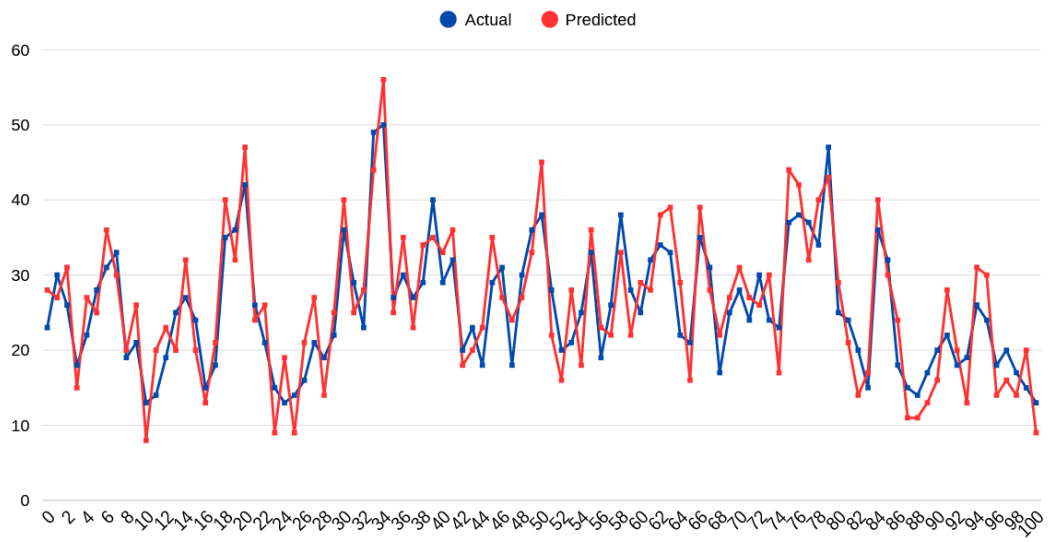

Fig 3. The comparison of predicted and actual values for the PM10 parameter

By incorporating both the variance of the observed values and the deviations of predictions from the mean, the Index of Agreement offers a balanced evaluation of how well the model reproduces observed patterns. As a result, it serves as a reliable indicator of the consistency and overall quality of forecasting results.

The mathematical formulation of the Index of Agreement is given in **Eq. (6)**. [12]

$$IA = \sum_{i=1}^{n} (O_i - P_i)^2 \Big/ \sum_{i=1}^{n} (|P_i - \bar{O}| + |O_i - \bar{O}|)^2, \quad (6)$$

where, O_i is the observation value, P_i is the predicted value, $\bar{O}$ is the average observation value, $\bar{P}$ is the average predicted value.

Root mean square error (RMSE) is a widely adopted metric for evaluating the accuracy of predictive models, measuring the deviation between predicted and observed values. It is calculated as the square root of the average of the squared differences between the model's predictions and the actual observations. This provides an aggregate measure of the magnitude of prediction errors.

By squaring the individual errors before averaging them, the RMSE places greater importance on larger deviations, making it particularly sensitive to significant prediction errors. This characteristic is useful in applications such as air quality forecasting, where large discrepancies may have critical implications.

RMSE offers a straightforward measure expressed in the same units as the target variable, facilitating direct comparison between predicted and observed values. Consequently, it serves as an effective indicator of the model's overall predictive performance. The formula for RMSE calculation is defined in **Eq. 7**.

$$RMSE = \sqrt{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 / n}\,, \quad (7)$$

where, y_i is the observation value, $\hat{y}_i$ is the predicted value, n is the number of observed values.

The coefficient of determination (CoD) R^2 is a widely used statistical measure that evaluates how well a regression model explains the variability in observed data. It represents the proportion of variance in the dependent variable that can be explained by the model. Values of R^2 range between 0 and 1; values closer to 1 indicate that the model successfully captures most of the underlying variability in the data, while values near 0 suggest limited explanatory power.

This metric provides insight into how well the predicted values align with actual observations by comparing residual errors with total data variation. As such, it is commonly used to evaluate the effectiveness and reliability of predictive models, particularly in regression-based forecasting tasks.

The coefficient of determination is useful for evaluating how well a model reproduces overall trends in the data, although it does not directly reflect the magnitude of prediction errors. Therefore, it is often used alongside other performance metrics to provide a more comprehensive evaluation. The formula for the CoD is defined in **Eq. 8**.

$$R^2 = 1 - \left(\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 / \sum_{i=1}^{n} (y_i - \bar{y}_i)^2 \right), \tag{8}$$

where, y_i is the observation value, $\hat{y}_i$ is the predicted value, $\bar{y}_i$ is the average value of y_i, n is the number of observed values.

Table 1 presents a comparative evaluation of three performance metrics – the Index of Agreement (IA), the CoD, and RMSE for the proposed LSTM-based model and several baseline approaches, including Support Vector Regression, XGBoost, Ridge Regression, and a Single-Factor Prediction model.

Table 1. **Comparison of the prediction results of the different models**

Model	IA	RMSE	CoD
Our model	0.92	11.7	0.77
Single-factor LSTM	0.88	26.14	0.63
XGBoost	0.37	67.94	0.75
Support Vector Regression	0.25	56.13	0.12
Ridge Regression	0.24	79.31	0.49

The results presented in **Table 1** demonstrate that the proposed Dual-Stream LSTM model provides the most balanced and accurate forecasting performance among the evaluated approaches. The improvement can be attributed to the ability of the model to simultaneously capture temporal dependencies in air quality measurements and the influence of meteorological conditions through the dual-stream architecture. Ridge Regression and Support Vector Regression exhibit substantially lower performance, which may be explained by their limited capacity to represent complex nonlinear relationships and long-term temporal patterns inherent in environmental data. XGBoost achieves a coefficient of determination comparable to that of the proposed model; however, the considerably lower Index of Agreement and higher RMSE values indicate reduced prediction consistency and larger forecasting errors. The comparison with the Single-Factor LSTM model demonstrates the contribution of meteorological information to the prediction process, as the integration of weather-related variables enables more comprehensive modeling of pollutant dynamics. The obtained results confirm the effectiveness of combining multi-source data with an attention-based fusion mechanism for air quality forecasting.

The model is trained using the Adam optimizer with a learning rate of 0.001. The sequence length (time step) is set to 24 to capture daily temporal patterns in air quality data. A batch size of 32 is used to ensure stable gradient updates.

The LSTM networks consist of 64 hidden units with a dropout rate of 0.2 to mitigate overfitting. The models are trained for 100 iterations. For the KNN imputation method, the number of nearest neighbors is set to $k = 5$, which provides a balance between local sensitivity and robustness.

Table 2 provides the training hyperparameters of the model.

Table 2. **Training the hyperparameters of the model**

Parameter	Value
Number of iterations	150
Time step	24
Batch size	32
Learning rate	0.001
Hidden units	64
k (for KNN imputation)	5

CONCLUSION

This paper presents a Dual-Stream LSTM framework for air quality forecasting that integrates information from monitoring stations and meteorological data sources. The main scientific contribution of the study lies in the combination of dual-stream temporal feature extraction with an attention-based fusion mechanism, enabling the model to dynamically determine the relative importance of different data sources during prediction. This approach allows more effective modeling of the complex interactions between air pollutant concentrations and environmental conditions than conventional single-source forecasting methods.

The experimental evaluation demonstrates that the proposed model achieves superior predictive performance compared with traditional machine learning approaches and single-factor prediction models. The results indicate that incorporating complementary information from meteorological variables significantly improves forecasting accuracy and model consistency, highlighting the benefits of multi-source learning for environmental prediction tasks.

The proposed framework can support intelligent air quality monitoring systems by providing more reliable forecasts for environmental management, public health protection, and urban planning. Furthermore, the developed methodology is not limited to air quality applications and may be adapted to other multivariate time-series forecasting problems involving heterogeneous data sources.

The proposed framework provides a flexible and efficient solution for modeling complex relationships in air quality data. By combining multi-source inputs with an adaptive fusion strategy, it offers improved predictive performance and can be extended to other environmental forecasting applications.

ACKNOWLEDGMENTS AND FUNDING SOURCES

This work was supported by the Ministry of Education and Science of Ukraine (Project No. 0125U001883).

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that they have no competing interests.

AUTHOR CONTRIBUTIONS

Conceptualization, [I.R., H.K.]; methodology, [I.R., H.K.]; investigation, [I.R., H.K.]; writing – original draft preparation, [I.R., H.K.]; writing – review and editing, [I.R., H.K.]; visualization, [I.R., H.K.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Kotlia, P., Pant, J., & Lohani, M. C. (2025, September). Design and Implementation of Statistical Time Series Forecasting Model for Air Quality Assessment. In *2025 6th International Conference on Electronics and Sustainable Communication Systems (ICESC)* (pp. 913-918). IEEE. https://doi.org/10.1109/icesc65114.2025.11212556

[2] Pandian, E., et al. (2025). Air quality forecasting for predictive analysis using machine learning. In *Proceedings of the 9th International Conference on Inventive Systems and Control (ICISC)* (pp. 187–192). IEEE. https://doi.org/10.1109/icisc65841.2025.11188992

[3] Potey, S., et al. (2025). Air quality prediction using machine learning: A comprehensive review. In *Proceedings of the International Conference on Cognitive Computing in Engineering, Communications, Sciences and Biomedical Health Informatics (IC3ECSBHI)* (pp. 1089–1093). IEEE. https://doi.org/10.1109/ic3ecsbhi63591.2025.10991003

[4] Borah, J., et al. (2024). Timezone-aware auto-regressive long short-term memory model for multipollutant prediction. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–9. https://doi.org/10.1109/tsmc.2024.3463960

[5] Alani, N. H. S., Chand, P., & Al-Rawi, M. (2025). A Two-Stage Machine Learning Framework for Air Quality Prediction in Hamilton, New Zealand. *Environments*, 12(9), 336. https://doi.org/10.3390/environments12090336

[6] Çelikten, H. (2026). A Multi-Output Deep Learning Framework for Simultaneous Forecasting of PM10 and Air Quality Index in High-Altitude Basins: A Case Study of Igdir, Türkiye. *Sustainability*, 18(8), 3883. https://doi.org/10.3390/su18083883

[7] Rudavskyi, I., & Klym, H. (2026). Air quality prediction based on long short-term memory prediction model. *Measuring Equipment and Metrology*, 87(1). https://doi.org/10.23939/istcmtm2026.01.026

[8] Zhang, X., Sun, Z., Zhou, Z., Jamali, S., & Liu, Y. (2022). Analysis and Dynamic Monitoring of Indoor Air Quality Based on Laser-Induced Breakdown Spectroscopy and Machine Learning. *Chemosensors*, 10(7), 259. https://doi.org/10.3390/chemosensors10070259

[9] Taştan, M. (2025). Machine Learning–Based Calibration and Performance Evaluation of Low-Cost Internet of Things Air Quality Sensors. *Sensors*, 25(10), 3183. https://doi.org/10.3390/s25103183

[9] Okorn, K., & Hannigan, M. (2021). Improving Air Pollutant Metal Oxide Sensor Quantification Practices through: An Exploration of Sensor Signal Normalization, Multi-Sensor and Universal Calibration Model Generation, and Physical Factors Such as Co-Location Duration and Sensor Age. *Atmosphere*, 12(5), 645. https://doi.org/10.3390/atmos12050645

[11] Zhou, H., Wang, T., Zhao, H., & Wang, Z. (2022). Updated prediction of air quality based on kalman-attention-LSTM network. *Sustainability*, *15*(1), 356. https://doi.org/10.3390/su15010356

[12] Ma, X., Chen, T., Ge, R., Xv, F., Cui, C., & Li, J. (2023). Prediction of PM2.5 Concentration Using Spatiotemporal Data with Machine Learning Models. *Atmosphere*, 14(10), 1517. https://doi.org/10.3390/atmos14101517

ДВОПОТОКОВА МОДЕЛЬ ДОВГОЇ КОРОТКОЧАСНОЇ ПАМ'ЯТІ ДЛЯ ПРОГНОЗУВАННЯ ЯКОСТІ ПОВІТРЯ З ВИКОРИСТАННЯМ СТАНЦІЙНИХ ТА ПОГОДНИХ ДАНИХ

Іван Рудавський, Галина Клим
Національний університет «Львівська політехніка»,
вул. Степана Бандери 12, 79013 м. Львів, Україна

АНОТАЦІЯ

Вступ. Забруднення повітря залишається серйозною проблемою для навколишнього середовища та здоров'я населення, що вимагає точних і надійних методів прогнозування для підтримки процесу прийняття рішень та розробки стратегій щодо зменшення негативного впливу. Зі зростанням доступності екологічних даних, що надходять зі станцій моніторингу та метеорологічних джерел, сучасні підходи на основі даних стали незамінними для моделювання складної динаміки якості повітря. Велике значення має розробка надійних архітектур глибокого навчання, здатних враховувати як часові залежності, так і зовнішні екологічні впливи.

Методи. У цьому дослідженні пропонується модель довгої короткочасної пам'яті (ДКЧП) із двома потоками для прогнозування якості повітря, яка обробляє дані станцій та метеорологічні дані за допомогою двох паралельних потоків. Застосовується конвеєр попередньої обробки, що включає підстановку відсутніх значень методом *k*-найближчих сусідів (*k*-НС) та нормалізацію вхідних ознак за стандартизованою *z*-оцінкою. Вихідні дані потоків ДКЧП інтегруються за допомогою механізму злиття на основі уваги, який адаптивно присвоює важливість кожному джерелу даних. Ефективність моделі порівнюється з декількома базовими підходами.

Результати. Експериментальні результати свідчать про те, що запропонована модель забезпечує вищу точність прогнозування та стабільність порівняно з традиційними методами машинного навчання та однофакторними моделями. Механізм об'єднання даних на основі уваги ефективно враховує динамічні взаємозв'язки між даними станцій та метеорологічними умовами, що дозволяє моделі адаптуватися до мінливих впливів навколишнього середовища. Оцінка за допомогою індексу узгодженості, коефіцієнта детермінації та середньоквадратичної похибки.

Висновки. Запропонована архітектура двопотокової ДКЧП забезпечує ефективне рішення для інтеграції неоднорідних екологічних даних у завданнях прогнозування якості повітря. Завдяки поєднанню методів структурованої попередньої обробки даних із стратегією адаптивного об'єднання даних модель підвищує ефективність та надійність прогнозування. Отримані результати підкреслюють потенціал підходів на основі багатоджерельного глибокого навчання для моніторингу навколишнього середовища та можуть бути застосовані в інших задачах прогнозування часових рядів.

Ключові слова: Прогнозування якості повітря; ДКЧП; модель з двома потоками; метод *k*-НС; стандартизована *z*-оцінка; сенсор.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
30 April, 2026	04 June, 2026	05 June, 2026	29 June, 2026

Electronics and information technologies, 2026, 34, 217–224
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC: 004.8

SPEECH SEPARATION BY MODIFIED DEEP NON-LINEAR FILTERING MODEL

Andrii Tsemko**, *Ivan Karbovnyk
Faculty of Electronics and Computer Technologies
Department of Radiophysics and Computer Technologies
Ivan Franko National University of Lviv,
107 Gen. Tarnavskoho St., Lviv, 79017, Ukraine
** Corresponding author e-mail: andrii.tsemko@lnu.edu.ua*

Tsemko A., & Karbovnyk I. (2026). Speech Separation by Modified Deep Non-Linear Filtering Model. *Electronics and Information Technologies*, 34, 217–224. https://doi.org/10.30970/eli.34.14

ABSTRACT

Background. Automatic Speech Recognition (ASR) systems in single-channel experimental setups perform poorly in real-world scenarios when several talkers are placed near a microphone. While for multi-channel setups, speech (or source) separation can be solved by algorithmic approaches such as beamforming or machine learning approaches using deep non-linear filtering models, there is no such monaural algorithm that can efficiently separate speech.

Materials and Methods. We focus on a successive model designed for multi-channel speech extraction with a steering mechanism called deep non-linear filtering and modify it for single-channel speech separation. This LSTM-based (LSTM - Long short-term memory) model without a steering mechanism demonstrates high-quality speech extraction from a fixed angle, which makes the model promising for single-channel separation. We modified such a model by appending a parallel output head that generates a second gain mask in addition to the existing one. We train the model on the Libri2Mix dataset in noisy conditions to separate speeches and extract them without background noise. Additionally, we investigate the influence of bidirectional LSTM layers versus the unidirectional version. As the DNLF (Deep non-linear filtering) model is built with bidirectional LSTM layers, it cannot be used for real-time separation processing.

Results and Discussion. Our trained DNLF-SS-UNI (Deep non-linear filtering speech separation unidirectional) model with unidirectional LSTM layers demonstrates 3.41 dB of SI-SDR (Scale-invariant signal-to-distortion ratio) and 5.34 dB of SI-SDRi (Scale-invariant signal-to-distortion ratio improvement) on test subset of Libri2Mix dataset with background noise, while its bidirectional version called DNLF-SS-BI (Deep non-linear filtering speech separation bidirectional) achieves 5.82 dB and 7.76 dB for SI-SDR and SI-SDRi, respectively. However, the bidirectional version of the model requires approximately 2.3 times more parameters than its unidirectional counterpart.

Conclusion. The use of bidirectional LSTM layers in speech separation models of architectures such as deep non-linear filtering is crucial, as a model with unidirectional LSTM layers instead shows a performance drop of approximately 2.4 dB for SI-SDR and SI-SDRi metrics.

Keywords: recurrent neural networks, speech separation, machine learning

BACKGROUND

The main aim of this research is to investigate the deep non-linear filtering model [1] for a single-channel speech separation task. Originally, this model was built for speech extraction and beamforming tasks in a multi-channel microphone setup. The model demonstrated significant results for speech extraction from a specified relative angle by additional input processed by a steering mechanism in a condition with 4 speakers talking simultaneously. It is designed to process an input of stacked matrices of calculated spectrograms from microphones and generate complex gain masks applied to a reference channel. The training framework processes the input complex stacked spectrograms of multi-talker audio scenarios to extract only a single specified person. In one version of the model, the desired speaker to extract is placed at a specified relative or fixed angle to the microphone setup. The trained model demonstrates high-quality separation/isolation of the target speaker, making it highly useful for real-world scenarios.

We gained insight from this relatively small model and decided to experiment with its architecture for a speech separation task focusing on the single-channel setup. While it is possible to add output head to the existing DNLF model and train it for speech separation, a multi-channel setup usually does not require machine learning approaches; since talkers are mostly placed at different angles to the microphone array, they can be efficiently separated by beamforming algorithms or even by the existing DNLF model applied iteratively. It is possible to run the DNLF model several times (many times as needed to extract all speakers) by specifying each time required angle of each speaker time, thereby allowing the existing model to work for a speech separation task. Therefore, we decided to focus on updating the model structure for a single-channel setup, modifying it to process only one complex spectrogram (real and imaginary parts) while generating two complex gain masks for each speaker.

Designing efficient neural network models for speech separation in a single-channel setup requires capturing and processing both long- and short-term audio dependencies. While classical convolutional neural networks (CNNs) demonstrate high performance in various pattern recognition tasks like image processing, their fundamental characteristic of focusing on local features makes them less effective for tasks such as speech separation.

As an efficient speech separation convolution-based model, Conv-TasNet [2] demonstrates a high-quality separation level. Instead of classical convolutional neural network layers, this model relies on temporal convolutional network (TCN) layers [3], which are designed to process input data with an increasing dilation factor. This allows the convolutional model to process both long- and short-term audio dependencies. While our investigation focuses on complex data representation and relies on the Short-Time Fourier Transform (STFT) for the feature preparation phase—and the inverse STFT (iSTFT) to convert the signal from the frequency domain back into the time domain—Conv-TasNet implements a fully trainable time-domain model using trainable encoder-decoder layers. These encoder-decoder layers use single Convolutional and Transposed Convolutional layers to convert the time-domain signal into a pseudo-spectrogram format and vice versa. These pseudo-spectrograms can be described as magnitude spectrograms of the input signal, as they consist of only a single matrix with a fixed number of frequency bins. The primary advantage of the Conv-TasNet model is that it is a fully convolutional model designed for time-domain processing which, when using causal convolutions, can be deployed for real-time speech separation.

Another fully-convolutional model that demonstrates even higher efficiency than Conv-TasNet is a model called SuDoRM-RF [4]. This model is designed in a similar way, using trainable encoder-decoder layers, but relies on U-Net-style convolutional blocks. These U-Net blocks utilize downsampling and upsampling operations with skip connections, which help the model capture both long- and short-term dependencies. This architecture results in a smaller model size with fewer floating-point operations while still yielding highly

competitive separation results in both clean and noisy conditions. Specifically, the SuDoRM-RF model uses only 2.6M parameters—which is roughly half that of Conv-TasNet (5.1M parameters)—and requires only 7.6 GFLOPs compared to Conv-TasNet's 10 GFLOPs. Despite this smaller footprint, it still demonstrates a high separation level, achieving 9.24 dB SI-SDRi compared to the 9.52 dB SI-SDRi of Conv-TasNet [5] for a clean speech mixture without background noise. Ultimately, our investigation focuses on real-world scenarios for speech separation in these types of noisy conditions.

The Deep Non-Linear Filtering model relies on recurrent neural network layers—specifically Long Short-Term Memory (LSTM) layers, which were originally designed to capture both long- and short-term dependencies. Another core difference is its use of an STFT signal representation instead of a trainable encoder layer. While a trainable encoder-decoder layer can potentially be trained to denoise a signal from non-speech components, the STFT provides a complex representation that supplies the model with both magnitude and phase components. This representation helps the model more clearly and precisely process the input information to reconstruct cleanly separated audio signals. Another important architectural feature is the processing of input matrices in both the wideband and narrowband dimensions [6]. This technique helps the model capture feature dependencies across both frequencies and time frames. Combined, these features allow the model to distinguish one speaker from another by focusing not just on isolated characteristics like pitch or talking speed, but on a combination of both.

MATERIALS AND METHODS

A real-world scenario for a speech separation task can be described by next equations:

$$Y(t) = \sum_{i}^{N} x_i(t) + n(t), \tag{1}$$

where $Y(t)$ is a time-domain representation of the mixture of N talkers defined as $x_i(t)$ where i is the index of a speaker and $n(t)$ is an additional background noise. In our particular case, we consider only a two-speaker scenario where N equals 2. The main task for our model is to process the input mixture and reconstruct $x_i(t)$ for each speaker. Our investigated model is designed to process input complex spectrograms as a tensor Y of shape [T, F, 2C], where T is a number of time frames, F number of frequency bins of STFT, and thirds dimension is equals to C by 2, where C is number of channels/microphones and 2 for both real and imaginary parts of spectrogram. In our particular case, C is equal to 1, so the input matrixes shape is [T, F, 2]. However, as original model was built for a single complex gain mask estimation of shape [T, F, 2], our output shape is [T, F, 2*N], so for our case we have an output shape of [T, F, 4], where first estimated gain mask Y_E_1 contains real and imaginary parts at third-dimension's index 0 and 1, and second estimated gain mask Y_E_2 contains real and imaginary parts at index 2 and 3, respectively.

$$Y_{re} = \mathrm{Re}[STFT(y)], \tag{2}$$

$$Y_{im} = \mathrm{Im}[STFT(y)], \tag{3}$$

The model architecture of the deep non-linear filtering model consists of 2 stacked LSTM layers, followed by a fully-connected layer with a tanh activation function. The overall model architecture is illustrated in **Figure 1**. The first LSTM layer, referred to as the

frequency-based LSTM (F-LSTM), processes the input complex spectrogram of a shape [T, F, 2] across the first dimension, so it obtains dependencies across frequencies (inter-frequency dependencies). To capture information across time frames, the resulting output is then permuted into a [F, T, N_F1] format, where N_F1 is the number of hidden features from the first LSTM layer. Due to this permutation, the second LSTM layer features dependencies (temporal dependencies) across time frames rather than frequencies. Through this dual-path approach, the model processes features along both the frequency and time dimensions. The output is then permuted back to a shape of [T, F, N_F2], where N_F2 is the number of features from the second LSTM layer. Next, the resulting matrices must be mapped back into real and imaginary components for our two target masks. To achieve this, we employ a fully-connected layer that processes each [T, F] slice from the [T, F, N_F2] tensor independently, mapping the N_F2 values to 4 output channels to form a vector of values: [$mask1_{real}$, $mask1_{imag}$, $mask2_{real}$, $mask2_{imag}$]. The obtained tensor of shape [T, F, 4] is then processed by a tanh activation function. Finally, the resulting complex masks are separated and applied to the input [T, F, 2] spectrogram independently to yield the individual spectrograms for each speaker, which are subsequently converted back into the time domain via the inverse STFT (iSTFT).

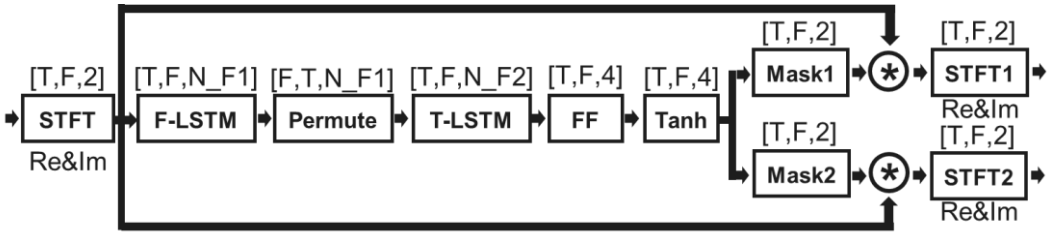

Fig. 1. DNLF-SS model architecture.

Since the original deep non-linear filtering model uses only bidirectional LSTM layers, it cannot be deployed in real-time applications. To address this, we decided to train two versions of the DNLF-SS model: one utilizing the original bidirectional LSTM layers and a second using unidirectional LSTM layers. A bidirectional LSTM layer use pair of weights and processes the input signal from the past-to-future (forward) and from the future-to-past (backward). While this allows the model to leverage future context, it prevents real-time execution because the model must wait for future frames that are unavailable in real-world streaming scenarios. In contrast, a unidirectional LSTM layer processes the sequence only in the forward direction. This makes real-time deployment possible, as the model can process the current time frame using only the current input and the hidden states from past frames [7]. Furthermore, switching to unidirectional LSTM layers reduces the parameter size of the recurrent block by half.

We utilized the open-source Libri2Mix dataset [8] with additional noise for training and evaluation. The dataset consists of mixtures of two speakers with background noise, where the Signal-to-Noise Ratio (SNR) ranges from -6 dB to +3 dB between the loudest speaker and the background noise. The audio data was segmented into 3-second frames at a 16 kHz sampling rate. We use the train-360 dataset subset for training and the test subset for evaluation. The network was trained for 100 epochs with 2 samples per batch. The Adam optimizer was used for training with an initial learning rate of 10^{-3}, which is reduced by a factor of 1.1 every 30 epochs.

As the loss function for training, we use the negative scale-invariant signal-to-distortion ratio (SI-SDR) [9]. The SI-SDR is calculated directly in the time domain rather than the frequency domain by comparing the target time-domain speech signals with the estimated ones. Because speech separation training focuses on obtaining outputs without a

predefined speaker order, we apply an utterance-level permutation-invariant training (uPIT) technique [10].

$$\alpha = \frac{\hat{x}^T s}{\|x\|^2}, \tag{4}$$

$$\text{SI-SDR}(x, \hat{x}) = 10 \log_{10} \left(\frac{\|\alpha x\|^2}{\|\alpha x - \hat{x}\|^2} \right), \tag{5}$$

where x is a target signal, and $\hat{x}$ is estimated speech.

$$L(X, \hat{X}) = -\max_{\pi \epsilon \Pi_N} \left[\frac{1}{N} \sum_{i=1}^{N} \text{SI-SDR}(x_i, \hat{x}_{\pi(i)}) \right], \tag{6}$$

where X is the list of N target sources, $\hat{X}$ is the list of N estimated sources, and the $\max[\,]$ operation ensures the optimal assignment.

In addition to SI-SDR, we also employ the SI-SDR improvement (SI-SDRi) metric for evaluation. This metric estimates the improvement in the separation level achieved by the model compared to a baseline. As a baseline, we calculate the initial SI-SDR metric for the original mixed signal

$$\text{SI-SDRi}(x, \hat{x}, mix) = \text{SI-SDR}(x, \hat{x}) - \text{SI-SDR}(mix, \hat{x}). \tag{7}$$

RESULTS AND DISCUSSION

We trained two LSTM-based models for speech separation in noisy conditions: DNLF-SS-UNI, which utilizes unidirectional LSTM layers, and DNLF-SS-BI, which utilizes bidirectional LSTM layers. The comparative results of our proposed models against the Conv-TasNet baseline are presented in Table 1.

Table 1. **Comparison for Libri2Mix test**

Models	SI-SDR (dB)	SI-SDRi (dB)	Params. (M)	FLOPs (G)
Conv-TasNet	4.48	6.41	5.1	9.992
DNLF-SS-UNI	3.41	5.34	1.4	0.013
DNLF-SS-BI	5.82	7.76	3.3	0.026

Our DNLF-SS-BI model demonstrates an improvement of 2.4 dB over the DNLF-SS-UNI variant across both the SI-SDR and SI-SDRi metrics. All results listed in the table were evaluated on the test subset of the Libri2Mix dataset under noisy background conditions. Although the bidirectional version of the model requires roughly 2.3 times more parameters than its unidirectional counterpart, it still maintains 1.8M fewer parameters than Conv-TasNet. Furthermore, both of our proposed models require fewer than 250M FLOPs to process 1 second of audio, whereas the fully-convolutional Conv-TasNet model requires approximately 10 GFLOPs.

CONCLUSION

In this study, we investigated a deep non-linear filtering model modified for single-channel speech separation. Our best model achieves a performance of 5.82 dB in the SI-SDR metric and 7.76 dB in SI-SDRi, outperforming the Conv-TasNet model by 1.34 dB across both metrics. Additionally, we demonstrate that the backward temporal direction is crucial for the separation performance of LSTM-based models. Utilizing unidirectional layers in the deep non-linear filtering model resulted in a performance drop of over 2.4 dB for each metric, while reducing the model size by only 1.9M parameters. Although bidirectional LSTM layers yield higher separation results and better quality in the reconstructed audio, their architecture prevents them from being used in online separation tasks because they require buffering at least 1 second of audio. In contrast, both Conv-TasNet and our proposed unidirectional model, DNLF-SS-UNI, support efficient real-time online processing.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that the research was conducted in the absence of any conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, [A.T., I.K.]; methodology, [A.T., I.K.]; validation, [I.K.]; formal analysis, [A.T., I.K.]; investigation, [A.T., I.K.]; writing – original draft preparation, [A.T.]; writing – review and editing, [I.K.]; visualization, [A.T.]; supervision, [I.K.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Tesch, K., & Gerkmann, T. (2023). Insights into Deep Non-linear Filters for Improved Multi-channel Speech Enhancement. *IEEE/ACM Transactions of Audio, Speech and Language Processing, vol 31.,* 563-575. https://doi.org/10.1109/TASLP.2022.3221046

[2] Luo, Y., & Mesgarani, N. (2019). Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation. *IEEE/ACM Trans. Audio, Speech, Language Process, 27(8),* 1256–1266. https://doi.org/10.1109/TASLP.2019.2915167.

[3] Lea, C., Vidal, R., Reiter, A., & Hager, G. D. (2016). Temporal Convolutional Networks: A Unified Approach to Action Segmentation. https://doi.org/10.48550/arXiv.1608.08242

[4] Efthymios., T., Zhepei., W., Paris., S. (2020). Sudo rm -rf: Efficient Networks for Universal Audio Source Separation. *IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*. https://doi.org/10.1109/MLSP49062.2020.9231900

[5] Tsemko, A., Karbovnyk, I. (2025). Edge-ready speech separation with Sudo-TasNet. *Advances in cyber-physical systems,* Vol. 10, No. 2. https://doi.org/10.23939/acps2025.02.202

[6] Chenda, L., Zhuo, C., Yi, C., Cong, H., Tianya, Z., Keisuke, K., Marc, D., Shinji, W., Yanmin, Q. (2021). Dual-Path Modeling for Long Recording Speech Separation in Meetings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 30,* 1508-1520. https://doi.org/10.1109/TASLP.2022.3165712

[7] Tsemko, A., Karbovnyk, I. (2024). Models and Methods for speech separation in digital systems. *Advances in cyber-physical systems*, Vol. 9, No. 2. https://doi.org/10.23939/acps2024.02.121
[8] Cosentino, J., Pariente, M., Cornell, S., Deleforge, A., & Vincent, E. (2020). LibriMix: An open-source dataset for generalizable speech separation. *ArXiv preprint.* arXiv:2005.11262. https://doi.org/10.48550/arXiv.2005.11262.
[9] Le Roux, J., Wisdom, S., Erdoğan, H., & Hershey, J. R. (2018). SDR – half-baked or well done? *ArXiv preprint*. arXiv:1811.02508. https://doi.org/10.48550/arXiv.1811.02508.
[10] Morten, K., Dong, Y., Zheng-Hua, T., & Jesper, J. (2017). Multitalker Speech Separation With Utterance-Level Permutation Invariant Training of Deep Recurrent Neural Networks. *IEEE/ACM Trans. Audio, Speech and Lang. Proc. 25(10),* 1901-1913. https://doi.org/10.1109/TASLP.2017.2726762.

РОЗДІЛЕННЯ МОВЛЕННЯ З ВИКОРИСТАННЯМ МОДИФІКОВАНОЇ ГЛИБИННОЇ МОДЕЛІ НЕЛІНІЙНОЇ ФІЛЬТРАЦІЇ

Андрій Цемко*, Іван Карбовник
Кафедра радіофізики та комп'ютерних технологій,
Львівський національний університет імені Івана Франка
вул. Ген. Тарнавського, 107, 79017 Львів, Україна

АНОТАЦІЯ

Вступ. Ефективність автоматизованих систем розпізнавання мовлення для одного мікрофона суттєво знижується в реальних умовах, коли одночасно говорять декілька людей поруч із пристроєм. Хоча для систем із багатьма мікрофонами задача розділення мовлення або розділення джерел сигналу загалом вирішується за допомогою таких алгоритмів, як адаптивне формування діаграми спрямованості (англ. beamforming), або з використанням нейронних мереж як глибокі моделі нелінійної фільтрації, ці підходи є неефективними для розділення мовлення в системах з одним мікрофоном.

Матеріали та методи. Ми обрали ефективну глибоку модель нелінійної фільтрації, розроблену для багатоканальних систем виділення мовлення зі стеринг-механізмом (керуванням діаграмою спрямованості), і модифікували її для одноканальної системи розділення мовлення. Ця модель в основі побудована з використанням так званих ДКЧП-шарів (ДКЧП – довга короткочасна пам'ять, англ. Long short-term memory, LSTM), однак без застосування поворотного механізму і показує високу ефективність для виділення одного мовця на фіксованому куті відносно лінійки мікрофонів, що наштовхнуло нас на ідею, що дана модель, перебудована для одноканальної системи, може бути також ефективно. Ми додали до цієї моделі додатковий вихід для генерації другої маски для другого мовця на додачу до вже існуючої. Використовуючи набір даних Libri2Mix з додатковим шумом ми навчили дану модель розділенню мовлення без фонових шумів. Також ми дослідили вплив використання саме дво-направлених ДКЧП шарів та односпрямованих ДКЧП шарів, оскільки оригінальна глибинна модель нелінійної фільтрації через використання дво-направлених шарів не може бути застосована для розділення мовлення у реальному часі.

Результати й обговорення. Модель ОН-ГНФ-РМ (ОН-ГНФ-РМ – одно-напрямлена глибока нелінійна фільтрація для розділення мовлення, англ. DNLF-SS-UNI – Deep non-linear filtering speech separation unidirectional) з одно-напрямленими ДКЧП шарами демонструє рівень розділення в 3,41 дБ для метрики MI-BCC

(масштабно-інваріантного відношення сигналу до спотворень, англ. SI-SDR – Scale-invariant signal-to-distortion ratio) та 5,34 дБ для пMI-BCC (покращення масштабно-інваріантне відношення сигналу до спотворень, англ. SI-SDRi – Scale-invariant signal-to-distortion ration improvement) на тестовому наборі даних з Libri2Mix з фоновими шумами, в той час як модель ДН-ГНФ-РМ (ДН-ГНФ-РМ – дно-напрямлена глибока нелінійна фільтрація для розділення мовлення, англ. DNLF-SS-BI - Deep non-linear filtering speech separation bidirectional) з дво-напрямленими ДКЧП шарами показує вище результати в 5,82 дБ та 7,76 дБ для метрик SI-SDR та SI-SDRi відповідно. Однак, дво-направлена версія моделі потребує майже в 2,3 рази більше навчальних параметрів, порівняно з одно-направленою версією.

Висновки. Використання дво-напрямлених ДКЧП шарів для задачі розділення мовлення в архітектурах по типу глибинної моделі нелінійної фільтрації є вирішальним, оскільки модель з однонапрямленими ДКЧП шарами показує зниження ефективності розділення на 2,4 дБ для обох метрик, – як MI-BCC, так і пMI-BCC.

Ключові слова: рекурентні нейронні мережі, розділення мовлення, машинне навчання

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
27 May, 2026	22 June, 2026	26 June, 2026	29 June, 2026

Electronics and Information Technologies, 2026, 34, 225–238
http://publications.lnu.edu.ua/collections/index.php/electronics/index

UDC 535.323, 535.53, 537.226, 548

ELASTIC PROPERTIES OF $RbNH_4SO_4$ CRYSTALS

***Oleh Markevych*[1], *Vasyl Stadnyk*[1], *Oleh Bovgyra*[1], *Roman Lys*[2]*, *Igor Matviishyn*[2], *Oleh Onufriv*[3]**
Ivan Franko National University of Lviv.
[1]Faculty of Physics, 19 Drahomanova St., Lviv, 79005, Ukraine
[2]Faculty of Electronics and Computer Technologies,
107 Henerala Tarnavskoho St., Lviv, 79017, Ukraine
[3]Ukrainian National Forestry University,
103 Henerala Chuprynky St., Lviv, 79057, Ukraine

Markevych, O., Stadnyk, V., Bovgyra, O., Lys, R., Matviishyn, I., & Onufriv, O. (2026). Elastic Properties of $RbNH_4SO_4$ Crystals. *Electronics and Information Technologies*, 34, 225–238. https://doi.org/10.30970/eli.34.15

ABSTRACT

Background. The elastic properties of a material are of great importance in solid-state physics, as they significantly influence the optical properties of crystals.

Materials and Methods. The elastic properties of rubidium ammonium sulfate ($RbNH_4SO_4$) crystals are investigated in this work. The calculation of elastic properties was carried out using the CASTEP program.

Results and Discussion. Analysis of the elastic coefficients matrix of $RbNH_4SO_4$ reveals that this crystal belongs to the rhombic system with pronounced elastic anisotropy. The crystal is mechanically stable, with the greatest rigidity along the *Y*–direction and the smallest along the *X*–direction. The most stable shear is in the *XZ* plane, and the least stable is in the *XY* plane. The calculation of the Cauchy relations revealed a significant deviation ($C_{ij} > C_{kl}$), indicating the predominance of non-central forces in interatomic interaction and a significant contribution of the covalent or ionic component of the bond in the $RbNH_4SO_4$ lattice.

The spatial distributions of Young's modulus, Young's modulus under universal compression, shear modulus, and Poisson's ratio were calculated. The degree of anisotropy in the elastic properties of the $RbNH_4SO_4$ crystal was estimated, confirming its significant anisotropy.

Conclusion. The Born criteria calculation showed that the crystal is mechanically stable. The calculation of the Cauchy relations revealed the predominance of non-central forces in the interatomic interaction and the presence of a significant contribution of the covalent or ionic component of the bond in the $RbNH_4SO_4$ lattice.

The presence of rigid tetrahedral groups $[SO]_4^{2-}$ causes the formation of covalent directed bonds within these groups. It is shown that the polarizability of ions is important: rubidium ions (Rb^+) and sulfate groups $[SO]_4^{2-}$ have large electron shells that deform during mechanical compression of the crystal. This leads to the formation of induced dipole moments. The complex nature of the forces in the $RbNH_4SO_4$ crystal is a consequence of the fact that the crystal consists not of point charges, but of polarizable ions and molecular complexes $[SO]_4^{2-}$ with their own internal covalent bond. This makes the material elastically anisotropic and sensitive to the direction of the applied load.

Keywords: crystal, anisotropy, elastic coefficients, Young's modulus, universal compression modulus, Poisson's ratio.

INTRODUCTION

The refractive indices n_i (RI) and birefringence Δn_i are among the most important optical parameters of materials in the spectral region of transparency. Along with their wide technical application in instrument-making and optoelectronics (information recording, transmission, and storage systems, spatial control of light beams, high-precision polarimetry devices), they are widely used as an effective tool for studying physical processes in crystals. In particular, they allow analyzing the features of electronic polarization, determining the contributions of various effects (thermo-optical, piezo-optical, electro-optical, acousto-optical, etc.) to changes in optical properties, estimating the magnitude of spontaneous polarization and anti-polarization, and investigating their changes during phase transitions. One of the newest applications of their application is biomedicine (for example, studying the microscopic structure of collagen fibers in connective tissue) [1-12].

This research allows us to study piezo- and electro-optical effects, electro- and piezogyration, spatial modulation phenomena, acousto-optic interaction, and photorefraction. It's noteworthy that the optical properties of crystals are closely related to their elastic characteristics. Deformations of the crystal lattice, which arise under the action of mechanical stresses, temperature, or external fields, lead to a change in electronic polarizability and, accordingly, to a change in the refractive indices. This is the basis for the piezo-optical and photoelastic effects, which establish a direct connection between the material's optical and elastic parameters. Therefore, for a thorough understanding of optical characteristics, it is necessary to consider the anisotropy and the values of the elastic constants of crystals, which determine the lattice's response to external influences [13-21].

This article considers the elastic properties of rubidium ammonium sulfate (RAS), $RbNH_4SO_4$ crystals. Previously, the temperature- and spectral-dependence of the main RIs in these crystals was studied. It was established that in the studied spectral region (300–800 nm) and temperature interval (77–820 K), the dispersion $n_i(\lambda)$ is normal and is well described by the two-oscillator Sellmeier formula. The intersection of the dispersion curves $n_i(\lambda)$ in the *Y*-direction was found. The refractions, electronic polarizabilities, and parameters of effective ultraviolet and infrared oscillators were calculated. A 2nd-order phase transition was observed at T = 120 K in the $RbNH_4SO_4$ crystal.

The band-energy structure of $RbNH_4SO_4$ crystals was calculated, and it was found that the top of the valence band of these crystals is localized at point D (**k** = {0.5; 0.5; 0}), the bottom of the conduction band is at point Γ (E = 5.16 eV), and the smallest direct band gap (point Γ) is 5.33. The top of the valence band is formed by the bonding *p*-orbitals of sulfur and oxygen. The bottom of the conduction band is mainly formed by the *s*- and *p*-states of NH_4 and Rb, hybridized with the antibonding *p* states of sulfur and oxygen. It was found that the band gap width of $RbNH_4SO_4$ crystals decreases with increasing pressure ($dE_g / d\sigma \sim -1.55 \cdot 10^{-5}$ eV/bar) [22-25].

Despite the considerable amount of available information regarding the optical and electronic properties of $RbNH_4SO_4$ crystals, their elastic characteristics remain practically unexplored. This lack of information significantly limits a comprehensive understanding of the coupling between mechanical deformation and optical response in these materials. Elastic properties are among the fundamental characteristics of crystalline solids because they determine the propagation of acoustic waves, mechanical stability, energy transfer processes, thermal transport, and the response of a crystal to external mechanical loading. In anisotropic crystals, the elastic response strongly depends on crystallographic direction, resulting in elastic anisotropy that plays a crucial role in many physical phenomena.

The study of elastic anisotropy has become particularly important in recent years due to the rapid development of acoustic metamaterials, phononic crystals, and anisotropic functional media. Modern materials science increasingly focuses on designing materials with highly anisotropic elastic properties, enabling unprecedented control over the

propagation of elastic and acoustic waves. Such materials can be employed for acoustic waveguiding, vibration suppression, sound insulation, acoustic cloaking, and advanced signal-processing systems. Elastic anisotropy also governs the directional dependence of sound velocities, acoustic attenuation, and acousto-optic interaction efficiency, making it a key parameter for the development of modern acoustic and photonic technologies.

From the viewpoint of crystal optics, elastic anisotropy is directly related to photoelastic, piezo-optic, and acousto-optic phenomena. Mechanical deformations modify the dielectric tensor of a crystal through the photoelastic effect, thereby changing its refractive indices and birefringence. Consequently, knowledge of the elastic tensor is indispensable for understanding and predicting the optical response of crystals under external stress. This is particularly important for materials that may serve as active elements of optical modulators, deflectors, polarization controllers, and optical sensing devices.

The mixed-cation nature of $RbNH_4SO_4$ crystals makes them especially interesting for elastic investigations. The coexistence of relatively large Rb^+ ions and orientationally active NH_4^+ groups creates a complex system of interatomic interactions, which may result in pronounced anisotropy of the elastic tensor. The presence of rigid $[SO]_4^{2-}$ tetrahedra combined with comparatively flexible cation sublattices is expected to generate anisotropic deformation mechanisms associated with both translational displacements of cations and rotational distortions of sulfate tetrahedra. Therefore, the analysis of elastic anisotropy can provide valuable insight into the microscopic nature of interatomic bonding and structural stability in these crystals.

In view of these considerations, the present work is devoted to a comprehensive investigation of the elastic properties of $RbNH_4SO_4$ crystals. Special attention is focused on the determination of the elastic stiffness coefficients, evaluation of the mechanical stability conditions, analysis of elastic anisotropy, and discussion of the relationship between elastic behavior, crystal structure, and optical characteristics. The obtained results are expected to contribute to a deeper understanding of the mechanical and acousto-optic properties of $RbNH_4SO_4$ crystals and to assess their potential for applications in modern optical and acoustic devices.

MATERIALS AND METHODS

The elastic properties of the $RbNH_4SO_4$ crystal were calculated using the CASTEP program, based on density functional theory (DFT) [26-32]. This technique combines the pseudopotential approach with plane waves, which makes it fast and widely used in solid-state chemistry and physics. The following electronic configurations were used for the valence electrons: O ($1s^2\ 2s^2\ 2p^4$); S ($1s^2\ 2s^2\ 2p^6\ 3s^2\ 3p^4$), N ($1s^2\ 2s^2\ 2p^3$), H ($1s^1$), Rb ($4s^2\ 4p^6\ 5s^1$). The ionic core was described using a norm-conserving pseudopotential [29], and the limiting kinetic energy controlling the number of plane waves was taken to be E_{cut} = 750 eV. Integration over the 1st Brillouin zone was performed at points of a 3×2×4 k-grid selected by the Monkhorst-Pack scheme [30].

To describe quantum exchange and correlation effects, the local density approximation (LDA) and the generalized gradient approximation (GGA) with the Perdew-Burke-Ernzerhof (PBE) parameterization were initially considered [31, 32]. As is well known, the use of the LDA functional typically leads to an underestimation of lattice parameters (by 1–3 %), resulting in a significant overestimation of elastic moduli and elastic coefficients C_{ij} (sometimes up to 10–20 % compared to experimental data). Conversely, the PBE functional tends to overestimate lattice parameters (by 1–2 %), which leads to a systematic underestimation of the elastic constants. Therefore, the PBEsol functional [33] was employed to ensure a precise balance between the cell geometry and the strain-energy relation. The PBEsol functional, specifically adapted for solids, provides a well-balanced description of the potential energy surface and yields elastic constants C_{ij} that lie between the LDA and PBE results. PBEsol is recognized as one of the most accurate and

well-balanced approximations for minimizing systematic errors in the calculation of elastic properties, as it delivers results comparable to those obtained from much more computationally expensive hybrid functionals, such as HSE and PBE0 [34,35].

The optimization procedure was performed similarly to works [36-38] with the following convergence criteria: maximum force $3 \cdot 10^{-2}$ eV/Å; maximum stress $5 \cdot 10^{-3}$ GPa; maximum atomic displacement $1 \cdot 10^{-4}$ Å. To calculate the elastic properties of the $RbNH_4SO_4$ crystal, known crystallographic data for this crystal with the space symmetry group No. 2 (P_{nam}) were used (**Table 1**). The results of the lengths and occupancy degrees of the shortest atomic bonds of this crystal are also given there.

Table 1. **Crystal lattice parameters, length, and degree of occupancy of the shortest atomic bonds in the rubidium-ammonium sulfate crystal**

Parameter	Experiment	Calculation
a, Å	7.830	7.696
b, Å	10.467	10.368
c, Å	5.977	5.932
V, Å^3	489.855	473.326
Z	4	4
Bond	**Length (Å)**	**Occupancy**
N–H	1.028–1,051	0.85–0.78
S–O	1.459–1,492	0.67–0.56
H–H	1.655–1,725	from –0.10 to –0.09
O–H	1.746–2,705	from 0.11 to –0.01
O–O	2.410–2,434	from –0.24 to –0.23
S–H	2.610–2,665	–0.03
N–O	2.796–2,935	from –0.14 to –0.04
Rb–O	2.867	0.08

The elastic properties are described by Hooke's law, which is written as follows for anisotropic materials [13, 39]:

$$\sigma_{mk} = C_{mknl}\varepsilon_{nl}, \tag{1}$$

where σ_{mk} is the stress tensor, C_{mknl} is the elastic coefficients, and ε_{nl} is the strain tensor. The $RbNH_4SO_4$ crystal belongs to the orthorhombic symmetry class. From a symmetry point of view, the elastic coefficients of such a crystal form a symmetric 6×6 matrix (in Voigt notation) containing 12 nonzero components, nine of which are independent: C_{11}, C_{22}, C_{33}, C_{44}, C_{55}, C_{66}, C_{12}, C_{13}, and C_{23}.

The coefficients C_{ij} were obtained by expanding the total energy $E(V, \delta)$ of the crystal in a Taylor series with respect to the unit cell volume V. For small deformations ($\delta \ll 1$), the energy per unit volume is described as follows [39]:

$$E(V, \delta) = E(V_0, 0) + V_0 \sum_{i=1}^{6} \sigma_i \, e_i + \frac{V_0}{2} \sum_{i,j=1}^{6} C_{ij} e_i e_j \,, \tag{2}$$

where V_0 is the equilibrium volume at zero pressure; e_i are the components of the deformation tensor in Voigt notation; C_{ij} are the second-order elastic coefficients, which are defined as the curvature of the energy surface:

$$C_{ij} = \frac{1}{V_0}\left[\frac{\partial^2 E}{\partial e_i \partial e_j}\right]. \quad (3)$$

Here, a set of strain tensors is used to calculate all 9 coefficients. Each strain δ causes an energy change $\Delta E = E(\delta) - E(0)$, which is a quadratic function of δ. Tensile/compressive strains along the axes are used to calculate the longitudinal and transverse moduli C_{ij}.

As an example, in the case of strain along the *X* axis ($e = \{\delta, 0, 0, 0, 0, 0\}$), it is obtained that:

$$\frac{\Delta E}{V_0} = \frac{1}{2} C_{11}\delta^2. \quad (4)$$

For orthogonal deformation (*XY* plane), we will have $e = \{\delta, -\delta, 0, 0, 0, 0\}$, which will correspond to the relation:

$$\frac{\Delta E}{V_0} = (C_{11} + C_{22} - 2C_{12})\delta^2/2. \quad (5)$$

Using the coefficients C_{11} and C_{22}, the coefficient C_{12} can be calculated. To calculate the shear moduli (C_{44}, C_{55}, C_{66}), shear deformations with volume conservation were used. In the case of shear in the *YZ* plane ($e = \{0, 0, 0, \delta, 0, 0\}$) we can write:

$$\frac{\Delta E}{V_0} = \frac{1}{2} C_{44}\delta^2. \quad (6)$$

Similar expressions can be written for the coefficients C_{55} (displacement in the *XZ* plane) and C_{66} (displacement in the *XY* plane). For calculations of the bulk modulus *B*, it is taken into account that the crystals of the rhombic syngony are related to the elastic moduli C_{ij} by the formula:

$$B = \frac{1}{9}\{C_{11} + C_{22} + C_{33} + 2(C_{12} + C_{13} + C_{23})\}. \quad (7)$$

Poisson's ratios (V_{ij}), which show the transverse compression of the crystal in the *j* direction during its stretching along the *i* direction, were calculated using the following formulas:

$$V_{XY} = -\frac{C_{11}}{C_{12}},\ V_{XZ} = -\frac{C_{11}}{C_{13}},\ V_{YZ} = -\frac{C_{22}}{C_{23}},$$
$$V_{YX} = -\frac{C_{22}}{C_{21}},\ V_{ZX} = -\frac{C_{33}}{C_{31}},\ V_{ZY} = -\frac{C_{33}}{C_{32}}. \quad (8)$$

Using the elastic coefficient matrix, the spatial distribution of Young's modulus is calculated in the work. For this, first, the inverse matrix of elastic compliance coefficients $[S] = [C]^{-1}$ is found. Then the Young's modulus E in the direction given by the unit vector $\mathbf{n} = [l_1, l_2, l_3]$ (where l_i are the direction cosines relative to the crystallographic axes) is calculated by the general formula:

$$\frac{1}{E} = \sum_{i=1}^{3}\sum_{j=1}^{3}\sum_{k=1}^{3}\sum_{l=1}^{3} S_{ijkl} l_i l_j l_k l_l. \tag{9}$$

RESULTS AND DISCUSSION

The matrix of elastic coefficients C_{ij} (GPa) of the $RbNH_4SO_4$ crystal at room temperature looks like

$$\mathbf{C} = \begin{bmatrix} 17.45 & 6.05 & 5.87 & 0 & 0 & 0 \\ 6.05 & 19.56 & 5.54 & 0 & 0 & 0 \\ 5.87 & 5.52 & 17.88 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5.24 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6.78 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4.86 \end{bmatrix}. \tag{10}$$

We see that the matrix with nine independent non-zero components is characteristic of rhombic symmetry. In the given case, an approximation to tetragonal symmetry is observed, since $C_{11} \approx C_{33}$ (17.45 ≈ 17.88 GPa) and $C_{44} \approx C_{66}$. However, the difference between C_{22} and the components $C_{12} \neq C_{13}$ confirms the rhombic structure.

The rubidium ammonium sulfate (RAS) crystal $RbNH_4SO_4$ belongs to the orthorhombic system (space group of type P_{nma}); therefore, its tensor of elastic coefficients contains 9 independent components C_{ij}, which are given in Eq. (10). Regarding the shear stiffness, it can be noted that the shear resistance is maximum in the *XZ* plane (C_{55} = 6.78 GPa) and minimum and almost isotropic in the *YZ* and *XY* planes ($C_{44} \approx C_{66}$).

For a stable crystal, the Born conditions must be fulfilled. For crystals of the rhombic system, the Born conditions of mechanical stability are written as follows:

$$\begin{aligned} &C_{11} > 0, C_{22} > 0, C_{33} > 0, C_{44} > 0, C_{55} > 0, C_{66} > 0; \\ &[C_{11} + C_{22} + C_{33} - 2(C_{12} + C_{13} + C_{23})] > 0; \\ &(C_{11} + C_{22} - 2C_{12}) > 0; \quad (C_{11} + C_{33} - 2C_{13}) > 0; \quad (C_{22} + C_{33} - 2C_{23}) > 0. \end{aligned} \tag{11}$$

The check showed that

$$\begin{aligned} &C_{11} + C_{22} - 2C_{12} = 17.45 + 19.56 - 2 \cdot 6.05 = 24.91 > 0; \\ &C_{11} + C_{33} - 2C_{13} = 17.45 + 17.88 - 2 \cdot 5.87 = 23.59 > 0; \\ &C_{22} + C_{33} - 2C_{23} = 19.56 + 17.88 - 2 \cdot 5.54 = 26.36 > 0; \\ &C_{11} + C_{22} + C_{33} + 2(C_{12} + C_{13} + C_{23}) = 54.89 + 2 \cdot 17.46 = 89.81 > 0. \end{aligned}$$

We see that all the criteria are met, so the crystal is mechanically stable. Let us compare the main elastic coefficients. Since $C_{22} > C_{33} > C_{11}$, the greatest stiffness is along the *b*-direction, while along the *a*-direction it is the least. That is, the *b*-direction is the stiffest, and the *a*-direction is the most compressible. This is a typical result for crystals of

the $ABSO_4$ double sulfate family (A and B = Li, K, Na, Cs, Rb, and NH_4), in which stiffer polyhedral SO_4 chains form along one axis. Since $C_{55} > C_{44} > C_{66}$, the most stable shear is in the *XZ* plane, while in the *XY* plane it is the least stable. This may be due to weaker interionic interactions between layers or to the orientational mobility of NH_4^+ groups.

An important indicator is the deviation from the Cauchy relations ($C_{12} = C_{66}$, $C_{13} = C_{55}$, $C_{23} = C_{44}$). From the matrix, we see that C_{12} (6.02) ≠ C_{66} (4.86) and C_{23} (5.54) ≠ C_{44} (5.24). A significant deviation ($C_{ij} > C_{kl}$) indicates the predominance of non-central forces in the interatomic interaction and the presence of a significant contribution of the covalent or ionic component of the bond in the $RbNH_4SO_4$ lattice.

Analysis of the matrix shows that the $RbNH_4SO_4$ crystal belongs to the rhombic syngony with pronounced elastic anisotropy (maximum stiffness along the *Y* axis). The inequality in the elastic constants indicates the complex nature of interatomic forces, which extend beyond simple central interactions. This "violation" means that the real interaction in rubidium ammonium sulfate is much more complex.

The reason for the “complex nature” of the forces in $RbNH_4SO_4$ is the presence of rigid tetrahedral groups $[SO]_4^{2-}$. The bonding within these groups is covalent, meaning there are directional bonds. Unlike ionic bonding, which acts equally in all directions, directional bonds resist changes in the angles between atoms, not just changes in distance. This creates “angular” or “multi-particle” forces that do not fit into the model of simple central springs.

The ions' polarizability also plays an important role. The rubidium ions Rb^+, ammonium ions NH_4^+, and, especially, the sulfate groups $[SO]_4^{2-}$ have large electron shells that deform during mechanical compression of the crystal. In the case of crystal compression, induced dipole moments arise. The “dipole-dipole” interaction depends on the orientation, which again adds an off-center component to the total lattice energy.

From the matrix, we also see that, since the Cauchy parameter $g=C_{12}-C_{44}=6.05-5.24>0$, this indicates the predominance of ionic character with pronounced polarizability. The crystal resists a change in volume (compression) more than a change in shape (shear). This is a consequence of the fact that interatomic forces “hold” the structure not only due to electrostatic attraction, but also due to the rigid geometry of sulfate groups. That is, the complex nature of the forces in the $RbNH_4SO_4$ crystal is a consequence of the fact that the crystal consists not of point charges, but of polarizable ions and molecular complexes $[SO]_4^{2-}$ with their own internal covalent bond. This makes the material elastically anisotropic and sensitive to the direction of the applied load.

For orthorhombic crystals, the maxima of Young's modulus are usually close to the crystallographic axes. If we compare the diagonal components of the stiffness tensor: $C_{22}>C_{33}>C_{11}$, it follows that the maximum Young's modulus is along the b axis, along the c axis it is slightly smaller, and along the a axis it is minimal. That is, the indicatrix of Young's modulus should be elongated along the b axis. Physically, this means that the crystal is stiffest in the b direction, and deformation occurs most easily along the a direction. Since the coefficients of the shear moduli satisfy the following relationship: $C_{55}>C_{44}>C_{66}$, this means that the lowest shear resistance is in the plane (ab). In a spatial surface, this manifests itself as a flattening of the indicatrix (**Fig. 1a**).

Poisson's ratio, *ν*, is one of the fundamental characteristics of the elastic properties of solids. It describes the mechanical coupling between longitudinal and transverse deformations of a load. If a uniaxial tensile stress *σ* is applied to a body, it elongates along the direction of the load and simultaneously compresses in the transverse directions. Poisson's ratio ν is given by the formula:

$$\nu = -\frac{\varepsilon_{\perp}}{\varepsilon_{\parallel}}, \tag{12}$$

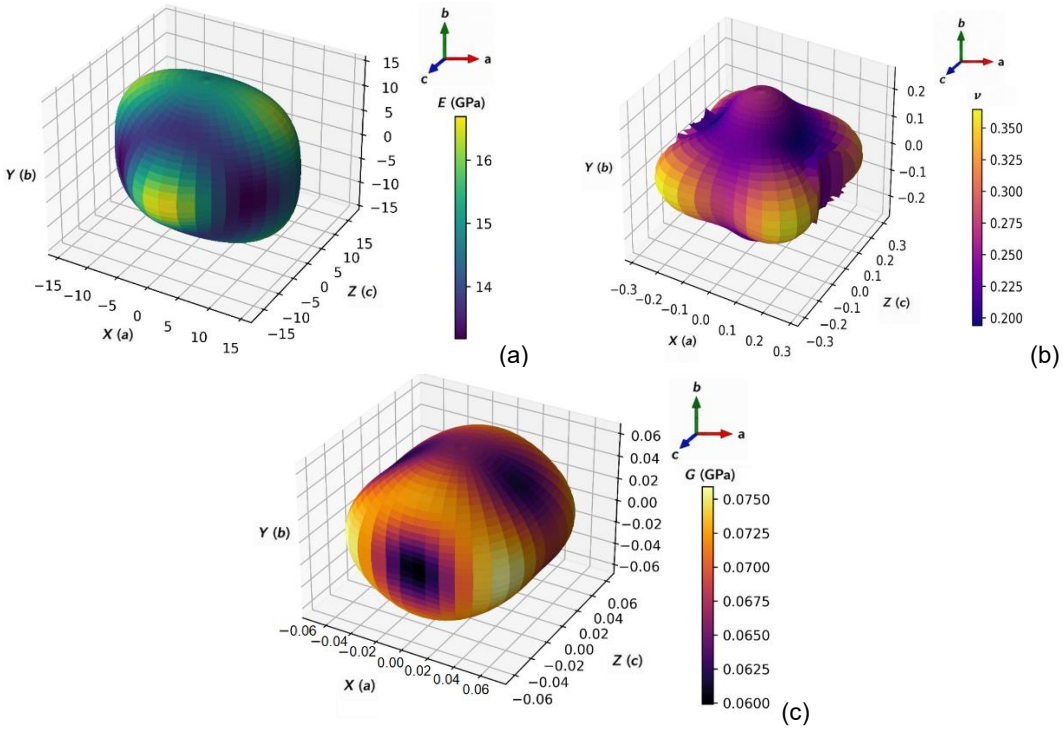

Fig. 1. Spatial distribution of Young's modulus (a), Poisson's ratio (b), and linear compressibility (c) of the $RbNH_4SO_4$ crystal at room temperature

where $\varepsilon_{\parallel}$ is the relative longitudinal strain, and $\varepsilon_{\perp}$ is the transverse strain. The minus sign in the formula means that the strains have opposite signs.

For most dielectric crystals, $\nu \approx 0.2–0.3$, which agrees well with the values obtained for the $RbNH_4SO_4$ crystal (**Fig. 1b**). In anisotropic crystals, ν is not a constant. It depends on the direction of the applied stress $\mathbf{n}$ and the direction of the transverse strain $\mathbf{m}$. Therefore, 3D Poisson's ratio surfaces or polar sections are constructed for crystals in certain crystallographic planes. In some directions, the value may differ significantly.

For an anisotropic crystal

$$\nu(\mathbf{n}, \mathbf{m}) = -\frac{s_{ijkl} n_i n_j m_k m_l}{s_{ijkl} n_i n_j n_k n_l}, \tag{13}$$

where s_{ijkl} is the compliance tensor, $\mathbf{n}$ is the load direction, and $\mathbf{m}$ is the direction of transverse deformation. This formula is used to construct spatial maps of ν.

For the $RbNH_4SO_4$ crystal, Poisson's ratio reflects the rigidity of the SO_4 tetrahedra, the relatively soft deformations of the NH_4^+ ion, and the weak participation of the Rb^+ ions in covalent bonding. This leads to moderate values of v and anisotropy of transverse deformations.

Poisson's ratio determines the propagation of acoustic waves, acousto-optic effects, the material's behavior under pressure, and the mechanical stability of the crystal. Therefore, its analysis is an important part of the study of the elastic and optical properties of crystals.

Poisson's ratio determines the transverse strain during tension. Since the off-diagonal coefficients C_{12}, C_{13}, and $C_{23} \approx 5.5–6.0$ GPa, are not significant enough, there is little transverse interaction between the axes in the crystal. Values of approximately $\nu \approx 0.18–$

0.30, maxima occur in directions close to the (*ab*) and (*ac*) planes, where shear strains are lighter.

Anisotropy in $RbNH_4SO_4$ is determined by its lattice structure: the rigid structural elements of the SO_4 tetrahedra are held together by strong covalent bonds. The softer elements, the NH_4^+ ion, form hydrogen bonds that are much softer. The weakly bound Rb^+ cation is mainly ionically bound. Therefore, there is great rigidity along the directions of the SO_4 chains, and there are lighter deformations in the directions involving NH_4.

The obtained surfaces (**Fig. 1c**) indicate moderate elastic anisotropy and have pronounced directions of light shear. This is important for piezo-optical effects, baric changes in refractive indices, and acousto-optical properties.

Compared to the isomorphic Rb_2SO_4 crystal, $RbNH_4SO_4$ usually exhibits lower shear moduli and higher anisotropy. The reason is the presence of the NH_4 ion, which reduces the coefficients due to orientational degrees of freedom. The main rigidity of the $RbNH_4SO_4$ crystal is formed by SO_4 tetrahedra. Such properties are in good agreement with phase transitions, anisotropy of thermal expansion, and anisotropy of thermo-optic coefficients dn/dT.

Thus, the $RbNH_4SO_4$ crystal has mechanical stability, moderate anisotropy of longitudinal moduli, significant anisotropy of shear deformations, and a relatively small bulk modulus (~10 GPa). Such properties are typical for ionic-molecular sulfate crystals with SO_4 tetrahedra and orientationally mobile NH_4 groups.

Young's modulus characterizes the rigidity of the material during uniaxial tension or compression. In crystals, the strain depends on the orientation of the load relative to the crystallographic axes. Therefore, the directional Young's modulus $E(\theta, \varphi)$ is introduced, where θ and φ are spherical angles that define the direction in the crystal. The Young's modulus was calculated using formula (9).

The dependence $E(\theta, \varphi)$ is usually depicted as a three-dimensional surface. In each direction, the radius r=$E(\theta, \varphi)$ is plotted. The resulting surface is called the Young's modulus indicatrix. Its shape characterizes the degree of elastic anisotropy and the directions of maximum and minimum stiffness. For isotropic materials, this surface is a sphere $E(\theta, \varphi)$=$const$, for weakly anisotropic crystals, it is close to an ellipsoid, and for strongly anisotropic crystals, elongated petal structures or directions with very small E may occur. In most crystals, the maximum of Young's modulus corresponds to the directions of the strongest chemical bonds or the densest packing of atoms. The directions of weak interionic bonds or possible slip planes correspond to the minimum value.

It is known that the structure of the $RbNH_4SO_4$ crystal is determined by SO_4 tetrahedra, Rb^+ cations, and NH_4^+ groups. The rigidity of the lattice is determined by strong covalent S–O bonds, ionic bonds with the Rb atom, and rather weak NH_4 hydrogen interactions. Therefore, Young's modulus has maxima along the directions where the SO_4 tetrahedra are more tightly connected, and minima in the directions where deformation occurs due to rotation or displacement of the NH_4 groups.

The Young's modulus is related to the bulk modulus B, the shear modulus G, and the Poisson's ratio ν by the following relationship (isotropic case): E=$9BG/(3B$+$G)$. Therefore, the spatial distribution of Young's modulus E is a combination of the shear stiffness and compressibility of the crystal. The shear anisotropy of crystals of the orthorhombic system is characterized by the following coefficients:

$$A_1 = \frac{4C_{44}}{C_{11} + C_{33} - 2C_{13}};\ A_2 = \frac{4C_{55}}{C_{11} + C_{22} - 2C_{12}};\ A_3 = \frac{4C_{66}}{C_{22} + C_{33} - 2C_{23}}. \quad (14)$$

If we substitute the values from Eq. (10) into these ratios, we can obtain the following values: $A_1 \approx 1.82$, $A_2 \approx 2.16$, and $A_3 \approx 1.03$. If $A = 1$, then there is isotropic behavior,

and if $A \neq 1$, then there is anisotropic behavior. Therefore, the greatest anisotropy is in the plane (*ac*), and the least is in the plane (*bc*).

Since $C_{22} > C_{33} > C_{11}$, we can estimate that $E_b > E_c > E_a$. That is, the maximum stiffness is along the b axis, the minimum along the a axis. The approximate ratio $E_{max}/E_{min} \approx 1.3 - 1.4$, which corresponds to moderate elastic anisotropy.

For the general characteristic, the parameter is used:

$$A^U = 5\frac{G_V}{G_R} + \frac{B_V}{B_R} - 6, \quad (15)$$

where G_V, B_V are Voith estimates, G_R, B_R are Reis estimates. For our set of constants, we obtain approximately $A^U \approx 0.4 - 0.6$. If $A = 0$, then there is practically no anisotropy, and if $A = 0..1$, then there is weak or moderate anisotropy. Thus, for the $RbNH_4SO_4$ crystal, there is moderate elastic anisotropy.

The structure of $RbNH_4SO_4$ contains rigid SO_4 tetrahedra, the Rb^+ ion, and NH_4^+ groups. The anisotropy arises from the different orientations of the SO_4 tetrahedra, the possibility of NH_4 rotation, and weaker hydrogen bonds. This leads to the fact that deformation occurs more easily in some planes and more difficultly along the directions of tight binding. For the Rb_2SO_4 crystal, the anisotropy is usually slightly smaller. The reason is the absence of the NH_4 molecular group and a more rigid ionic lattice.

CONCLUSION

The elastic properties of $RbNH_4SO_4$ crystals at room temperature have been investigated for the first time using a comprehensive analysis of the elastic stiffness tensor. The obtained results fill an existing gap in literature concerning the mechanical characteristics of this crystal and provide the necessary basis for understanding the coupling between its elastic, optical, and electronic subsystems.

Analysis of the elastic stiffness matrix confirmed that $RbNH_4SO_4$ belongs to the orthorhombic crystal system and exhibits pronounced elastic anisotropy. The inequality of the principal elastic constants ($C_{22} > C_{33} > C_{11}$) indicates that the crystal possesses the highest resistance to uniaxial deformation along the *Y* crystallophysical direction, whereas the *X* direction is mechanically the softest. A substantial anisotropy was also found for shear deformation, with the highest shear stability observed in the *XZ* plane and the lowest in the *XY* plane. These results reveal a strong directional dependence of the interatomic interactions and deformation mechanisms operating in the crystal lattice.

Mechanical stability was confirmed by satisfying all Born stability criteria. At the same time, a significant deviation from the Cauchy relations was established, providing clear evidence for the predominance of non-central interatomic forces. This behavior originates from the coexistence of rigid covalently bonded sulfate tetrahedra and comparatively flexible ionic sublattices formed by Rb^+ and NH_4^+ ions. The obtained results demonstrate that the elastic response of $RbNH_4SO_4$ cannot be described within the framework of simple central-force models and is governed by complex many-body interactions, directional bonding effects, and the intrinsic polarizability of the constituent structural units.

A detailed analysis of the elastic anisotropy revealed that the rigid $[SO]_4^{2-}$ tetrahedra act as the main structural elements controlling the mechanical behavior of the crystal. Under external loading, deformation is primarily accommodated through rotations and distortions of sulfate tetrahedra as well as displacements of Rb^+ and NH_4^+ ions. Such a deformation mechanism gives rise to pronounced anisotropy of the elastic tensor and explains the observed directional dependence of the Young's modulus, shear modulus, compressibility, and Poisson's ratio.

For the first time, three-dimensional spatial distributions of the Young's modulus, linear compressibility, shear modulus, and Poisson's ratio were calculated and analyzed. The resulting surfaces reveal a highly anisotropic mechanical response, which is directly related to the crystal structure and anisotropic arrangement of sulfate groups. The universal elastic anisotropy index was estimated to be $A^U \approx 0.4–0.6$, indicating a moderate-to-high degree of elastic anisotropy. The relatively small difference between the Voigt and Reuss approximations confirms the reliability of the elastic characteristics obtained while simultaneously reflecting the intrinsic anisotropic nature of the crystal.

The results obtained are important not only from a fundamental perspective but also for potential practical applications. Knowledge of the elastic tensor and elastic anisotropy is essential for evaluating the acousto-optic, photoelastic, and piezo-optic properties of $RbNH_4SO_4$ crystals. The pronounced anisotropy of the elastic response suggests the possibility of directional control of acoustic-wave propagation and acousto-optic interaction efficiency. Therefore, $RbNH_4SO_4$ can be considered a promising material for acousto-optic modulators, deflectors, polarization-control elements, and optical sensors operating under external mechanical loading.

Furthermore, the calculated elastic parameters provide a basis for future investigations of acoustic-wave velocities, photoelastic coefficients, acousto-optic figures of merit, and strain-induced modifications of the electronic band structure and refractive characteristics. Consequently, the present study establishes a fundamental framework for the development of multifunctional optical and acoustic devices based on $RbNH_4SO_4$ crystals.

ACKNOWLEDGMENTS AND FUNDING SOURCES

The authors received no financial support for the research, writing, and/or publication of this article.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that they have no competing interests.

AUTHOR CONTRIBUTIONS

Conceptualization, [O.M., V.S., O.B.]; methodology, [O.M., O.B., V.S.]; investigation, [O.M., V.S., R.L., I.M., O.O.]; writing – original draft preparation, [O.M., R.L.]; writing – review and editing, [O.M., V.S., O.B., R.L.]; visualization, [O.M., O.B.].

All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Goodman, J. W. (2005). *Introduction to Fourier optics*. Roberts & Company Publishers, 491 p.
[2] Flowers, G. R. (2012). *Introduction to modern optics*. Courier Corporation, 352p.
[3] Iizuka, K. (2002). *Elements of photonics: Volume II: For fiber and integrated optics*. John Wiley & Sons, 1222 p. https://doi.org/10.1002/0471221376
[4] Kubby, J. A. (2013). *Adaptive optics for biological imaging*. CRC Press, 388 p. https://doi.org/10.1201/b14898
[5] Goldstein, D. H. (2011). *Polarized light* (3rd ed.). CRC Press, 808 p. https://doi.org/10.1201/b10436
[6] Hao, R., Zeng, N., Zhang, Z., He, H., He, C., & Ma, H. (2024). Polarization feature fusion and calculation of birefringence dynamics in complex anisotropic media. *Optics Letters, 49*(9), 2273. https://doi.org/10.1364/OL.515983
[7] Zhang, Z., Hao, R., Shao, C., Mi, C., He, H., et al. (2023). Analysis and optimization of aberration induced by oblique incidence for in vivo tissue polarimetry. *Optics Letters, 48*(23), 6136–6139. https://doi.org/10.1364/OL.501365

[8] Ghosh, N., & Vitkin, A. (2011). Tissue polarimetry: Concepts, challenges, and applications. *Journal of Biomedical Optics, 16*(11), 110801. https://doi.org/10.1117/1.3652896
[9] Ramella-Roman, J. C., Saytashev, I., & Piccini, M. (2020). A review of polarization-based imaging technologies for clinical and preclinical applications. *Journal of Optics, 22*(12), 123001. https://doi.org/10.1088/2040-8986/abbf8a
[10] Zaffar, M., & Pradhan, A. (2020). Assessment of anisotropy of collagen structures through spatial frequencies of Mueller matrix images for cervical pre-cancer detection. *Applied Optics, 59*(4), 1237–1248. https://doi.org/10.1364/AO.377105
[11] Yang, H., Song, J., Zeng, N., & Ma, H. (2023). Local optimized Stokes polarimetry for specific polarization states. *Optics Letters, 48*(11), 3019–3022. https://doi.org/10.1364/OL.490110
[12] Wang, L., Peng, Y., Zhang, H., & Zhao, C. (2025). Generalized Stokes polarimetry design for a 3D polarization field. *Optics Letters, 50*(2), 293–296. https://doi.org/10.1364/OL.546036
[13] Nye, J. F. (1985). *Physical properties of crystals: Their representation by tensors*. Oxford University Press, 352 p.
[14] Ting, T. C. T. (1996). *Anisotropic elasticity: Theory and applications*. Oxford University Press, 592 p. https://doi.org/10.1093/oso/9780195074475.001.0001
[15] Newnham, R. E. (2005). *Properties of materials: Anisotropy, symmetry, structure*. Oxford University Press, 378 p. https://doi.org/10.1093/oso/9780198520757.001.0001
[16] Waxler, R. M., & Feldman, A. (1980). Optical properties of materials. *Applied Optics, 19*(15), 2481–2489. https://doi.org/10.1364/AO.19.002481
[17] Mytsyk, B. (2003). Methods for the studies of piezooptic effect in crystals and analysis of the experimental data. II. Analysis of the experimental data. *Ukrainian Journal of Physical Optics*, 4(3), 105–110. https://doi.org/10.3116/16091833/4/3/105/2003
[18] Oseledchik, S., Prosvirnin, A., Pisarevskiy, A. I., & others. (1995). Optical properties of anisotropic materials. *Optical Materials, 4*(6), 669–675. https://doi.org/10.1016/0925-3467(95)00027-5
[19] Polian, A., & Grimsditch, M. (1999). Elastic properties under pressure. *Physical Review B, 60*(3), 1468–1473. https://doi.org/10.1103/PhysRevB.60.1468
[20] Chilla, E., Flannery, C. M., Frohlich, H.-J., Straube, U., & others. (2001). Optical studies of crystals. *Journal of Applied Physics, 90*(12), 6084–6088. https://doi.org/10.1063/1.1409574
[21] Zhang, N., Wu, J., & Zhang, H. (2025). Crystal structure and optical properties. *Crystals, 15*(3), 269. https://doi.org/10.3390/cryst15030269
[22] Stadnyk, V. Y., Romanyuk, M. O., & Brezvin, R. S. (1997). Optical and electronic parameters of $RbNH_4SO_4$ crystals. *Ferroelectrics, 192*(1–4), 203–207. https://doi.org/10.1080/00150199708216190
[23] Andriyevsky, B., Ciepluch-Trojanek, W., Stadnyk, V., Tuzyak, M., Romanyuk, M., & Kurlyak, V. (2007). Band structure and optical spectra of $RbNH_4SO_4$ crystals. *Journal of Physics and Chemistry of Solids, 68*(10), 1892–1896. https://doi.org/10.1016/j.jpcs.2007.05.017
[24] Stadnyk, V. Y. (1998). Baric changes of optical indicatrix of $RbNH_4SO_4$ crystals [in Ukrainian]. *Ukrainian Journal of Physics, 43*(2), 213–218.
[25] Stadnyk, V., Vyshnevskyi, V., Matviishyn, I., Baliha, V., & Lys, R. (2025). Refractometry of uniaxially compressed Rb_2SO_4 crystals. *Electronics and Information Technologies*,30, 177–186. https://doi.org/10.30970/eli.30.14
[26] Clark, S. J., Segall, M. D., Pickard, C. J., Hasnip, P. J., Probert, M. I. J., Refson, K., & Payne, M. C. (2005). First principles methods using CASTEP. *Zeitschrift für*

Kristallographie – Crystalline Materials, 220(5–6), 567–570. https://doi.org/10.1524/zkri.220.5.567.65075
[27] Hohenberg, P., & Kohn, W. (1964). Inhomogeneous electron gas. *Physical Review, 136*(3B), B864–B871. https://doi.org/10.1103/PhysRev.136.B864
[28] Kohn, W., & Sham, L. J. (1965). Self-consistent equations including exchange and correlation effects. *Physical Review, 140*(4A), A1133–A1138. https://doi.org/10.1103/PhysRev.140.A1133
[29] Hamann, D. R., Schlüter, M., & Chiang, C. (1979). Norm-conserving pseudopotentials. *Physical Review Letters, 43*(20), 1494–1497. https://doi.org/10.1103/PhysRevLett.43.1494
[30] Monkhorst, H. J., & Pack, J. D. (1976). Special points for Brillouin-zone integrations. *Physical Review B, 13*(12), 5188–5192. https://doi.org/10.1103/PhysRevB.13.5188
[31] Perdew, J. P., Burke, K., & Ernzerhof, M. (1996). Generalized gradient approximation made simple. *Physical Review Letters, 77*(18), 3865–3868. https://doi.org/10.1103/PhysRevLett.77.3865
[32] Pfrommer, B. G., Côté, M., Louie, S. G., & Cohen, M. L. (1997). Relaxation of crystals with the quasi-Newton method. *Journal of Computational Physics, 131*(1), 233–240. https://doi.org/10.1006/jcph.1996.5612
[33] Perdew, J. P., Ruzsinszky, A., Csonka, G. I., Vydrov, O. A., Scuseria, G. E., Constantin, L. A., Zhou, X., & Burke, K. (2008). Restoring the density-gradient expansion for exchange in solids surfaces. *Physical Review Letters*, *100*(13), 136406. https://doi.org/10.1103/PhysRevLett.100.136406
[34] Råsander, M., & Moram, A. M. (2015). On the accuracy of commonly used density functional approximations in determining the elastic constants of insulators and semiconductors. *The Journal of Chemical Physics*, *143*(14), 144104. https://doi.org/10.1063/1.4932334
[35] Yuk, S. F., Sargin, I., Meyer, N., Krogel, J. T., Beckman, S. P., & Cooper, V. R. (2024). Putting error bars on density functional theory. *Scientific Reports*, *14*(1), 20219. https://doi.org/10.1038/s41598-024-69194-w
[36] Andrievsky, B., Jaskolski, M., Stadnyk, V., & others. (2013). Electronic structure calculations. *Computational Materials Science, 77*, 442–448. https://doi.org/10.1016/j.commatsci.2013.06.048
[37] Stadnyk, V., Gaba, V., Andrievskii, B., & Kogut, Z. (2011). Birefringence properties of mechanically clamped K_2ZnCl_4 crystals. *Physica Status Solidi B, 248*(1), 131–137. https://doi.org/10.1134/S106378341101029X
[38] Andrievsky, B., Romanyuk, M., & Stadnyk, V. (2009). Optical properties of materials. *Journal of Physics and Chemistry of Solids, 70*(7), 1109–1113. https://doi.org/10.1016/j.jpcs.2009.06.007
[39] Narasimhamurty, T. S. (1981). *Photoelastic and electro-optic properties of crystals*. Springer. https://doi.org/10.1007/978-1-4757-0025-1

ПРУЖНІ ВЛАСТИВОСТІ КРИСТАЛІВ $RbNH_4SO_4$

Олег Маркевич[1], Василь Стадник[1], Олег Бовгира[1], Роман Лис[2]*, Ігор Матвіїшин[2], Олег Онуфрів[3]
Львівський національний університет імені Івана Франка,
[1]Фізичний факультет, вул. Драгоманова 19, м. Львів, 79005, Україна,
[2]Факультет електроніки та комп'ютерних технологій,
вул. Генерала Тарнавського 107, м. Львів, 79017, Україна.
[3]Національний лісотехнічний університет,
вул. Генерала Чупринки 103, м. Львів, 79057, Україна

АНОТАЦІЯ

Вступ. Пружні властивості матеріалу є важливими для фізики твердого тіла, оскільки вони суттєво впливають на оптичні властивості кристалів.

Матеріали та методи. У роботі досліджено пружні властивості кристалів рубідій амоній сульфату $RbNH_4SO_4$. Розрахунок пружних властивостей проведено за допомогою програми CASTEP.

Результати. Аналіз матриці пружних коефіцієнтів $RbNH_4SO_4$ свідчить, що цей кристал належить до ромбічної сингонії із вираженою пружною анізотропією. Нерівність пружних констант вказує на складний характер міжатомних сил. Встановлено, що кристал є механічно стабільним, найбільша жорсткість є вздовж *Y* – напряму, найменша – вздовж *X*. Оскільки $C_{55} > C_{44} > C_{66}$, то найбільш стійкий зсув є в площині *XZ*, найменший – в площині *XY*. Розрахунок відношень Коші виявив значне відхилення ($C_{ij} > C_{kl}$), що вказує на переважання нецентральних сил у міжатомній взаємодії та наявність суттєвого внеску ковалентної або іонної складової зв'язку в ґратці $RbNH_4SO_4$.

У роботі розраховано просторовий розподіл модуля Юнга, модулі всебічного стискання, Юнга, зсуву та коефіцієнт Пуассона. Оцінено ступінь анізотропії пружних властивостей кристала $RbNH_4SO_4$, що підтверджує його значну анізотропію.

Висновки. Розрахунок критеріїв Борна показав, що кристал є механічно стабільним. Порівняння головних пружних коефіцієнтів показало, що найбільша жорсткість є вздовж *Y* – напряму, тоді як вздовж *X* – напряму вона є найменшою. Розрахунок відношень Коші виявив переважання нецентральних сил у міжатомній взаємодії та наявність суттєвого внеску ковалентної або іонної складової зв'язку в ґратці $RbNH_4SO_4$.

Наявність жорстких тетраедричних груп $[SO]_4^{2-}$ спричиняє утворенню ковалентниих напрямлених зв'язків всередині цих груп. Показано, що важливим є вплив поляризовності іонів: іони рубідію (Rb^+) та сульфатні групи $[SO]_4^{2-}$ мають велику електронну оболонку, яка деформується під час механічного стиснення кристалу. Це спричиняє виникнення індукованих дипольних моментів. Складний характер сил у кристалі $RbNH_4SO_4$ є наслідком того, що кристал складається не з точкових зарядів, а з поляризовних іонів та молекулярних комплексів $[SO]_4^{2-}$ із власним внутрішнім ковалентним зв'язком. Це робить матеріал пружно-анізотропним і чутливим до напрямку прикладеного навантаження.

Ключові слова: кристал, анізотропія, пружні коефіцієнти, модуль Юнга, модуль всебічного стискання, коефіцієнт Пуассона.

Received / Одержано	Revised / Доопрацьовано	Accepted / Прийнято	Published / Опубліковано
28 April, 2026	10 June, 2026	12 June, 2026	29 June, 2026

Збірник наукових праць

Електроніка та інформаційні технології

Electronics and information technologies

Випуск 34

2026

Підп. до друку 30.06.2026. Формат 70x100,16. Папір друк.
Друк на різогр. Гарнітура Times New Roman. Умовн. друк. арк. .
Тираж 100 прим. Зам. № .

Львівський національний університет імені Івана Франка.
79000 Львів, вул. Університетська, 1.

Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру видавців, виготівників і розповсюджувачів видавничої продукції. Серія ДК № 3059 від 13.12.2007 р.