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:

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

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:

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

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 |·| 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.

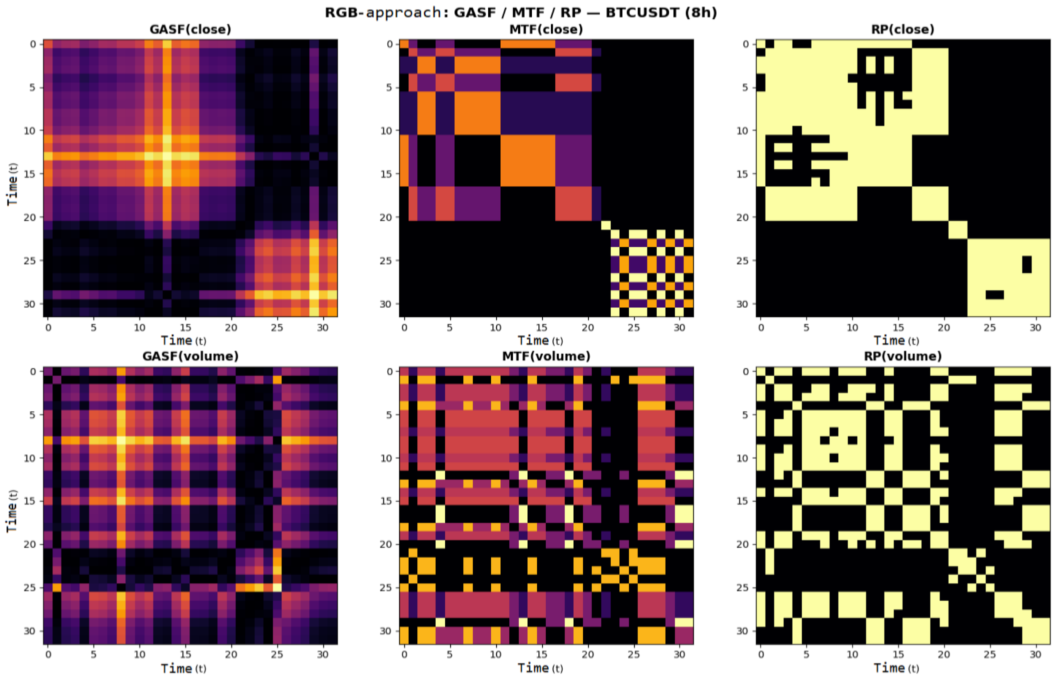

Fig. 1. RGB-approach of GASF, MTF and RP.

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