

УДК 004.8

<http://dx.doi.org/10.30970/vam.2024.33.12384>

ПРО МОДЕЛЬ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ОБЛИЧ МЕТОДАМИ КОМП'ЮТЕРНОГО ЗОРУ ТА МАШИННОГО НАВЧАННЯ

Ю. Юрченко¹, Н. Юрченко², А. Музичук¹

¹Львівський національний університет імені Івана Франка,
вул. Університетська 1, Львів, 79000, Україна
e-mail: yuliana.yurchenko1673@gmail.com, anatoliy.muzychuk@lnu.edu.ua,

²Ужгородський національний університет,
вул. Університетська 14, Ужгород, Україна
e-mail: nataliia.yurchenko@uzhnu.edu.ua

Розглянуто завдання класифікації зображень облич у реальному часі за певними ознаками. Зокрема, робота сфокусована на класифікації зображень облич за наявністю медичної маски. Оскільки обсяг анотованих наборів зображень людей в масках у відкритому доступі досить малий, ми запропонували підхід до генерування відповідних зображень на основі ключових точок обличчя (facial landmarks). Використання синтетичних даних для тренування класифікатора носіння маски призвело до збільшення його точності.

Розглядаючи актуальне завдання автоматизації контролю за дотриманням маскового режиму в громадських місцях, ми створили модель виявлення медичних масок на обличчях людей, яка поєднує модель виявлення облич YOLOv7Face та натренований класифікатор носіння маски. Для тестування кінцевої моделі зібрано набір зображень учнів і вчителів у навчальних закладах під час спалаху респіраторних захворювань. Результати обчислювальних експериментів підтвердили, що точність моделі досягає 87.2% у метриці mAP (mean Average Precision) на зібраному наборі зображень. Також з'ясовано, що використання моделі поліпшення якості зображень облич CodeFormer перед етапом їх класифікації зменшує похибку моделі виявлення масок, але суттєво збільшує час обробки вхідних зображень.

Ключові слова: комп'ютерний зір, згорткова нейронна мережа, виявлення об'єктів зображення, виявлення облич на зображенні, ключові точки обличчя, виявлення медичних масок на зображенні, обробка в реальному часі.

1. ВСТУП

Поєднання методів комп'ютерного зору та машинного навчання (ML – machine learning) зумовлюють бурхливий розвиток сучасних технологій обробки цифрових зображень та їхнє застосування у різних сферах. Зокрема це стосується автоматизації процесів виявлення людських облич на зображеннях та їхньої класифікації за певними ознаками в реальному часі.

Історію розвитку зазначених процесів можна умовно поділити на два основні етапи, відповідно до і після 2014 року [25], розглядаючи класичне виявлення об'єктів та їхнє виявлення на основі глибокого навчання. На першому етапі визначальними виявилися праці Віюлі та Джонса [22,23], у яких вперше було запропоновано систему виявлення людських облич у реальному часі, засновану на ідеях інтегрального зображення, вибору ознак і каскадів виявлення. Запропонований у праці Далала і Тріггса [4] алгоритм використання гістограм орієнтованих градієнтів (HOG – histogram of oriented gradients) значно поліпшив обчислення масштабно-інваріантних ознак, тобто ознак, які зберігають свої властивості зі зміною масштабу зображення.

Другий етап характеризується впровадженням глибоких згорткових нейронних мереж (DCNN – deep convolutional neural network), які здатні автоматично виділяти ознаки різних рівнів складності, починаючи з простих, таких як контури чи текстур, і завершуючи складними, що охоплюють семантичні концепції об'єктів. Це дає змогу моделі зберігати стійкість до змін освітлення, масштабів і спотворень зображення. Відповідні системи виявлення об'єктів можна умовно поділити на дві основні категорії. До першої належать двоетапні детектори, що послідовно локалізують і класифікують об'єкти, тут популярними є архітектури нейронних мереж на основі R-CNN (Region-Based Convolutional Neural Network) [9]. Іншу категорію становлять одноетапні детектори, наприклад, YOLO (You Only Look Once) [17], Single Shot MultiBox Detector (SSD) [12], RetinaNet [11].

Відомо, що ефективність кожної з моделей ML залежить від якості наборів даних, використаних для їхнього навчання та валідації. Зокрема, такі набори мають бути характерними для конкретної сфери застосування систем виявлення об'єктів на зображеннях. Наприклад, для аналізу зображень обличч з медичною маскою актуальними є набори зображень, анотованих стосовно наявності маски. Зазначимо, що для такої задачі є популярними набори Mask Dataset [13] і Face Mask Dataset [5]. Очевидно, що для класифікації зображень з іншими характерними ознаками, наприклад, окулярами, головними уборами, татуюваннями тощо треба використовувати інші спеціалізовані набори даних.

Важливою характеристикою набору даних є також його обсяг, який має бути достатньо великим. Зазначимо, що таку вимогу не завжди легко задовольнити, розглядаючи наявні джерела зображень. Тому в таких випадках актуальними стають синтетичні набори даних. Далі буде запропоновано підхід до генерування відповідних зображень, оперуючи ключовими точками обличчя (facial landmarks). Ефективність такого підходу буде продемонстровано для виявлення та класифікації зображень обличч з медичною маскою, що є актуальною проблемою стосовно забезпечення автоматизованого контролю за дотриманням маскового режиму в громадських місцях. Серед досліджень такої проблеми можна виділити працю [6], де запропоновано модель RetinaFaceMask для одноетапного виявлення одягнених медичних масок, а також публікацію [18], в якій розглянуто побудову двоетапної моделі для виявлення масок на обличчях, де для локалізації обличч використовується R-CNN, а для класифікації носіння маски тестуються кілька класифікаторів. Зазначимо, що у цих працях не висвітлено здатності моделей працювати в реальному часі.

Отже є актуальною розробка моделі для контролю у реальному часі за обличчями, які характеризуються певними ознаками – елементами зображення, що стосуються ключових точок обличчя. Простим, але важливим прикладом таких ознак можуть бути медичні маски, тому розроблений підхід продемонструємо для цього випадку.

2. ДЕТЕКТОРИ ОБ'ЄКТІВ НА ОБЛИЧЧЯХ

Для опрацювання цифрових зображень побудовано два детектори. Перший детектор двоетапний (див. рис.1). Він має стандартну архітектуру і виконує такі кроки:

- 1) виявляє обличчя на отриманому цифровому зображенні;
- 2) класифікує виявлене обличчя стосовно наявності маски.

На першому етапі детектор виявляє обличчя на цифровому зображенні за допомогою моделі YOLOv7Face з відкритим кодом [24]. На другому відбувається класи-

фікація виявлених облич на два класи: обличчя з масками та обличчя без масок, відповідно. Для класифікації була навчена модель на основі згорткової нейронної мережі EfficientNetV2-B3 [19].

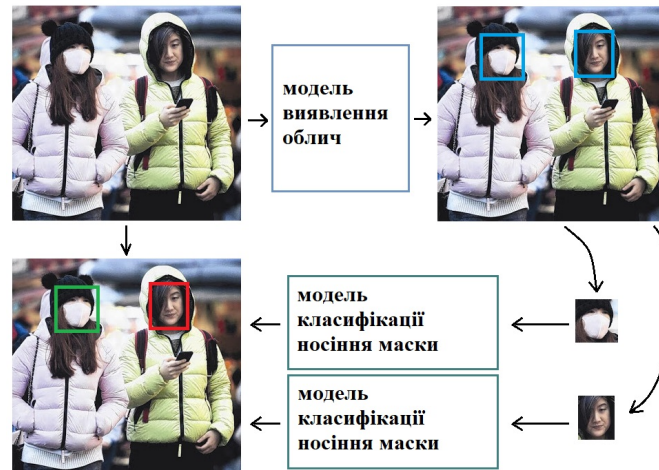


Рис. 1. Схема застосування до вхідного зображення двоетапної системи виявлення медичних масок на обличчях людей
Scheme of applying a two-stage system for detecting medical masks on people's faces to an input image

Зауважимо, що на практиці цифрові зображення бувають різної якості, що суттєво впливає на результати роботи детектора. Тому другий детектор розроблений як триетапний (див. рис. 2). Він відрізняється від базового двоетапного додатковим етапом, який після виявлення обличчя поліпшує якість зображень облич за допомогою CodeFormer [20]. Такий препроцесинг потребує додаткових обчислювальних ресурсів, але, як далі буде продемонстровано результатами апробації, дає змогу надійніше контролювати дотримання маскового режиму.

3. РОЗРОБКА СИНТЕТИЧНОГО НАБОРУ ЗОБРАЖЕНЬ

Як було вже зазначено, кількість доступних наборів зображень осіб у масках є значно меншою за обсягом від кількості зображень осіб без масок. Ця обставина зумовлює необхідність генерування синтетичних даних для успішного навчання і тестування автоматизованих систем контролю.

Розглянемо запропонований підхід для підготовки потрібних даних, беручи оригінальні зображення облич з набору VGGFace2 [3], що містить близько 3.3 мільйона зображень понад 9 тисяч осіб. Нехай маємо випадково вибране цифрове зображення людини з набору VGGFace2 та знайдені координати ключових точок обличчя. Приклад виявлення цих точок для конкретного обличчя показано на рис. 4. Фігура, яка окреслена контуром підборіддя та охоплює ніс, має подібну до медичної маски форму (див. рис. 5). Для створення синтетичних зображень осіб у масках ці фігури зафарбовують за допомогою функції `fillPoly` бібліотеки `OpenCV` однотонно одним з кольорів, як продемонстровано на рис. 6.

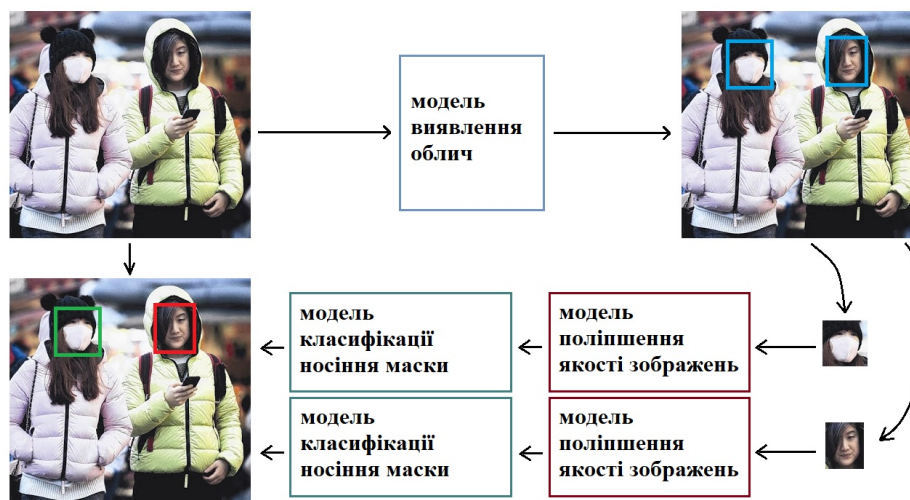


Рис. 2. Схема застосування до вхідного зображення триетапної системи виявлення медичних масок на обличчях людей
 Scheme of applying a three-stage system for detecting medical masks on people's faces to an input image

Як і в детекторах, виявлення обличчя на цифровому зображенні виконано за допомогою моделі YOLOv7Face. Для визначення ключових точок обличчя у цій праці використано метод SPIGA (Shape Preserving Facial Landmarks with Graph Attention Networks) [16]. Алгоритм SPIGA використовує згорткову нейронну мережу, а також каскад графових нейронних мереж з механізмом уваги, який дає змогу зберігати геометричну структуру обличчя, зберігаючи високу точність локалізації точок навіть зі зміною освітлення або виразу обличчя.

Зауважимо, що наведений алгоритм застосовний і для генерування зображень обличчя з іншими характеристиками (наприклад, наявними окулярами, капелюхом, татуванням тощо), якщо вони асоціюються з ключовими точками обличчя.

Для апробації запропонованого підходу випадково вибрано 18 тисяч зображень людей без масок з VGGFace2 і на їх основі створено 18 тисяч зображень осіб у масках. Отриманий у такий спосіб новий набір з 36 тисяч зображень було розподілено на тренувальну, валідаційну та тестову вибірки у співвідношенні 16 : 1 : 1, відповідно. Цей набір даних використано для навчання моделі класифікації з метою підвищення ефективності виявлення носіння масок.

Ефективність розроблених детекторів масок досліджували на спеціально сформованому наборі цифрових зображень учнів і вчителів у навчальних закладах [1]. Зображення для такого набору були завантажені з вебресурсів стокової фотографії Getty Images [8], Unsplash [21] та Pexels [15]. Вибрані зображення є мінімальних розмірів і такими, що могли б бути отримані з ракурсу камер відеоспостереження у закладах освіти. Набір містить 100 цифрових зображень різних розмірів, на кожному з яких наявна одна або кілька осіб. Створено анотацію, що описує обмежувальні прямокутники обличчя людей на зображеннях і відносить кожне з них до одного з двох класів: маска одягнена або маска не одягнена. Підготовлений у



Рис. 3. Виявлення обличчя
Face detection



Рис. 4. Виявлення ключових
точок обличчя
Facial keypoint detection

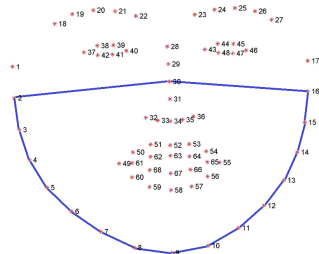


Рис. 5. Побудова фігури, що імітує маску
Building a shape that imitates a mask

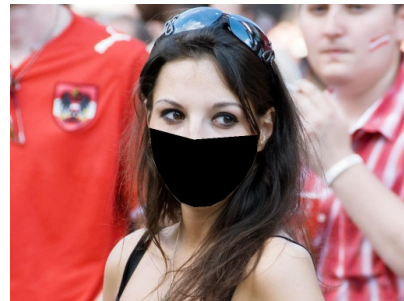


Рис. 6. Зображення з накладеною
маскою
Image with mask applied

Рис. 7. Схематичне відображення кроків алгоритму
накладання масок на зображення облич
Schematic representation of the steps of the algorithm
for applying masks to face images

такий спосіб набір зображень був використаний для тестування розробленої системи виявлення медичних масок на обличчях людей.

4. ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

Розглянутий підхід щодо виявлення заданих характеристик на зображенні обличчя (у цьому випадку медичної маски) реалізовано як прототип відповідної комп'ютерної системи в середовищах розробки PyCharm та JupyterLab мовою програмування Python 3 з використанням бібліотек Numpy, Matplotlib, OpenCV, Keras, TensorFlow, PyTorch, ptfllops [7], object_detection_metrics [14]; проєктів yolov7-face [24], CodeFormer [20], SPIGA [16], Keras_FLOP_Estimator [10]; а також платформи Docker для розгортання програмного забезпечення.

Завантаження та попередня обробка зображень, що використовувалися для тренування та валідації класифікатора наявності маски, реалізовані через клас ImageDataGenerator бібліотеки Keras. Для всіх зображень проводиться однаковий

етап попередньої обробки, який полягає у приведенні розміру зображення до розміру 64×64 пікселів (`target_size = (64, 64)`).

До тренувальної частини набору зображень також додатково застосовані такі перетворення:

- випадкове дзеркальне відображення стосовно вертикальної осі (`horizontal_flip=True`);
- масштабування на випадкове значення в межах заданого діапазону (`zoom_range=0.2`);
- випадковий зсув у межах визначеного інтервалу (`shear_range=0.2`);
- зміна яскравості шляхом множення на випадкове значення в межах діапазону (`brightness_range=[0.6, 1.0]`);
- Гауссове розмивання з випадковим розміром ядра, що є непарним числом у діапазоні $[0, 9)$.

У цій праці класифікатор носіння маски було побудовано за допомогою класу `Sequential` бібліотеки Keras у вигляді послідовного з'єднання таких елементів:

1. шарів моделі `EfficientNetV2-B3` (без останнього повнозв'язного шару);
2. шару зміни розмірності, який перетворює багатовимірний вектор у одномірний; цей шар реалізовано через клас `Flatten` з бібліотеки Keras;
3. вихідного повнозв'язного шару, що реалізовано класом `Dense` з бібліотеки Keras, з розмірністю 2 та нормованою експоненційною функцією активації (`activation=softmax`).

Інтерфейс класу `EfficientNetV2B3` з бібліотеки Keras дає змогу обрати початкові параметри (ваги) нейронної мережі `EfficientNetV2-B3`, що були отримані під час попереднього навчання на наборі даних ImageNet для класифікації зображень (`weights="imagenet"`). Аргумент `include_top=False` відкидає останній повнозв'язний шар моделі. Натомість додано новий вихідний повнозв'язний шар з розмірністю 2, для якого виконано оптимізацію параметрів; тренування інших шарів було вимкнено. Використання попередньо натренованих шарів для створення класифікатора носіння маски є ефективним з погляду скорочення часу на тренування та для підвищення точності передбачень.

5. МЕТРИКИ ДЛЯ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ

Для оцінювання результатів розпізнавання об'єктів використано метрику mean Average Precision (mAP) з пороговим значенням IoU, який дорівнює 0.5. Для обчислення цієї метрики було використано відкриту бібліотеку `object_detection_metrics` [14].

Для перевірки можливості роботи моделі в режимі реального часу виміряно кількість виконаних арифметичних операцій з плаваючою крапкою (FLOPs). Обчислення FLOPs для обробки тестових даних нейронними мережами було проведено за допомогою бібліотеки `ptflops` [7] (для моделей, реалізованих у PyTorch), а також проєкту [10] (для моделей, реалізованих у Keras). Для оцінки реального часу роботи моделі на конкретному процесорі важливо враховувати його тип та інші особливості. Наприклад, центральні процесори 12-го покоління Intel Core здатні виконувати кілька сотень GFLOPs на секунду [2], графічні процесори Nvidia GeForce RTX 30 Series можуть виконувати кілька TFLOPs на секунду.

6. РЕЗУЛЬТАТИ АПРОБАЦІЇ ЗАПРОПОНОВАНОГО ПІДХОДУ

Для аналізу впливу штучно створених тренувальних даних були навчені дві версії класифікатора носіння масок. Обидві нейронні мережі мають архітектуру EfficientNetV2-B3. Перший класифікатор Classifier-Orig-Mask навчався на суто оригінальних зображеннях облич із набору Face Mask Dataset протягом 6-ти епох. Класифікатор Classifier-Aug-Mask навчався на наборі зображень зі штучно доданими масками протягом 6-ти епох, а потім ще 3 епохи на оригінальному наборі зображень облич. Статистики точностей класифікацій протягом тренувального процесу зображено на рис. 8 і 9 для обох класифікаторів, відповідно.

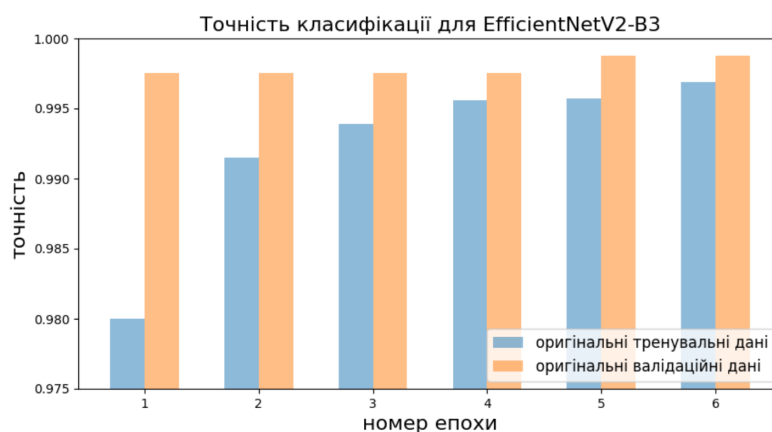


Рис. 8. Значення точності Classifier-Orig-Mask для тренувальної та валідаційної частин набору зображень облич
Classifier-Orig-Mask accuracy values for the training and validation parts of the face image set

Для дослідження також було поставлено мету оцінити вплив методів поліпшення якості зображень і порівняти ефективність двоетапного та триетапного підходів щодо виявлення масок на обличчях. У цьому розділі наведено статистичні результати для чотирьох версій детектора:

1. Detector-Orig – двоетапний детектор, що використовує Classifier-Orig-Mask.
2. Detector-Aug – двоетапний детектор, що використовує Classifier-Aug-Mask.
3. Detector-SR-Orig – триетапний детектор із поліпшенням якості зображень (SR – super-resolution), що використовує Classifier-Orig-Mask.
4. Detector-SR-Aug – триетапний детектор із поліпшенням якості зображень, що використовує Classifier-Aug-Mask.

Для тестування обрано 2 набори зображень: набір зображень учнів і вчителів у закладах освіти (позначимо його Dataset1) та Mask Dataset (позначимо Dataset2). У табл. 1 наведені значення точності класифікаторів за метрикою mAP, які обчислені шляхом порівняння анотованих і передбачених обмежувальних прямокутників для облич та відповідних передбачених ймовірностей носіння маски.

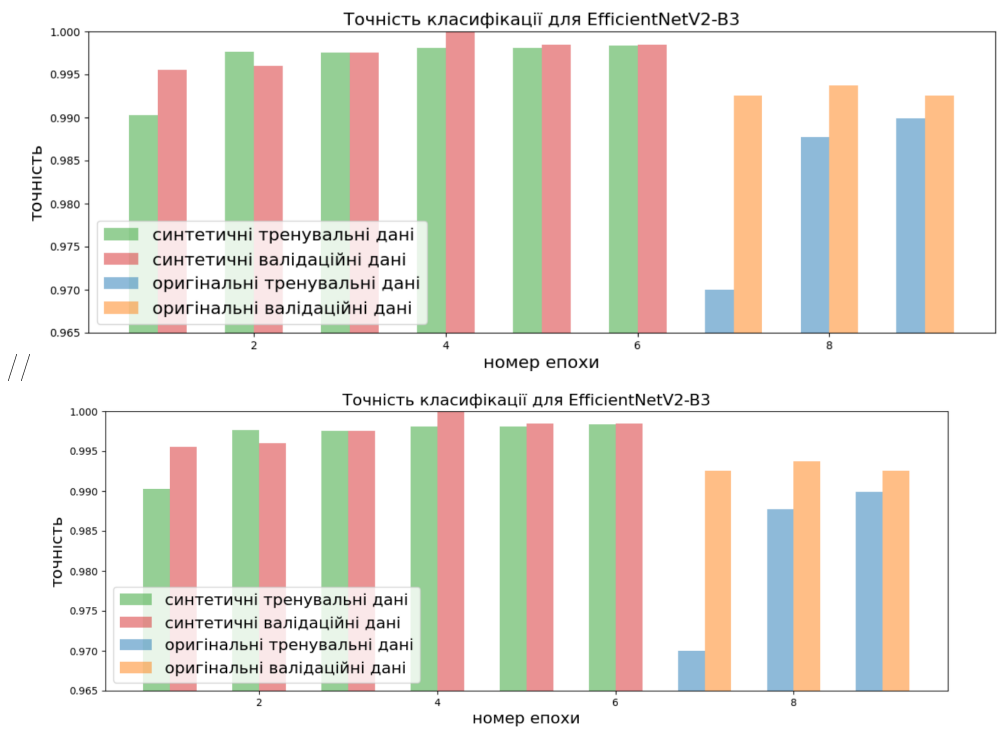


Рис. 9. Значення точності Classifier-Aug-Mask для тренувальних і валідаційних частин наборів зображень обличч
Classifier-Aug-Mask accuracy values for training and validation parts of face image sets

Таблиця 1

Значення mAP для результатів виявлення медичних масок на обличчях людей
mAP value for detection results of medical masks on people's faces

Версія детектора	mAP (%)	
	Dataset1	Dataset2
Detector-Orig	86.3	77.2
Detector-Aug	87.2	81.3
Detector-SR-Orig	87.3	78.2
Detector-SR-Aug	88.9	82.2

З метою оцінювання продуктивності розробленої моделі для виявлення масок було виміряно час її виконання на комп'ютері з процесором Intel Core i5-10300H та 16 ГБ оперативної пам'яті. У режимі тестування зображення обробляли послідовно. Отримані значення часу записано до табл. 2.

Кількість операцій з плаваючою крапкою, необхідних для обробки одного зображення, було обчислено програмно:

- yolov7-lite-s потребує ≈ 2.96 GFLOPs для обробки одного зображення розміру 640×640 ;
- EfficientNetV2-B3 потребує ≈ 0.27 GFLOPs для обробки одного зображення розміру 64×64 .

Таблиця 2

Значення середніх часу та кількості операцій з плаваючою крапкою,
необхідних для обробки одного зображення

Average time and number of floating point operations
required to process a single image

Версія детектора	Обчислювальна складність (GFLOPs)		Час (мілісекунди)	
	Dataset1	Dataset2	Dataset1	Dataset2
Detector-Orig	4.1	4.2	223.8	235.3
Detector-Aug	4.1	4.2	219.3	229.8
Detector-SR-Orig	-	-	17628.1	19451.0
Detector-SR-Aug	-	-	17901.5	19509.4

Для обчислення кількості FLOPs у системі виявлення медичних масок варто також врахувати, що класифікатор маски може бути застосований кілька разів до одного зображення, бо на ньому може бути кілька облич. Конкретно, набір Mask Dataset містить 853 зображення та близько 3800 виявлених облич, а набір учнів та вчителів – 100 зображень та 415 виявлених облич. Усі значення кількості FLOPs записано до табл. 2.

7. ВИСНОВКИ

Результати, отримані під час апробації запропонованого підходу до класифікації зображень облич стосовно носіння маски, демонструють його високу ефективність. А саме розширення наборів даних завдяки згенерованим згідно з запропонованим алгоритмом синтетичних зображень поліпшує точність класифікації зображень. Розроблений двоетапний детектор здатний працювати в режимі реального часу, обробляючи 4-5 зображень на секунду на процесорі Intel Core i5-10300H. Отримані значення точності у метриці mAP 81.3% для набору Mask Dataset та 87.2% для набору зображень учнів і вчителів демонструють високу ефективність запропонованого підходу до автоматизації контролю за дотриманням маскового режиму. Заразом триетапний детектор зменшує похибку класифікації в середньому на 8%, але в кілька разів збільшує час обробки вхідних даних порівняно з двоетапним. Тому на практиці його використання потребуватиме потужнішої обчислювальної техніки.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Набір зображень учнів та вчителів (Aug 2024) [Online]. Available: https://drive.google.com/drive/folders/13ZLZOJVmtijp_Ocj15upsYujCMA8v_v0?usp=drive_link
2. APP Metrics for Intel Microprocessors (Aug 2024) [Online]. Available: <https://www.intel.com/content/dam/support/us/en/documents/processors/APP-for-Intel-Core-Processors.pdf>

3. Cao Q. VGGFace2: A dataset for recognising faces across pose and age / Q. Cao, L. Shen, W. Xie, O. M. Parkhi, A. Zisserman // IEEE Conference on Automatic Face and Gesture Recognition. – 2018. – doi:10.48550/arXiv.1710.08092.
4. Dalal N. Histograms of oriented gradients for human detection / N. Dalal, B. Triggs // Computer Vision and Pattern Recognition. – 2005. – Vol. 1.
5. Face Mask Detection 12K Images Dataset (Aug 2024) [Online]. Available: <https://www.kaggle.com/datasets/ashishjangra27/face-mask-12k-images-dataset>
6. Fan X. RetinaFaceMask: A Single Stage Face Mask Detector for Assisting Control of the COVID-19 Pandemic / X. Fan, M. Jiang // IEEE International Conference on Systems, Man, and Cybernetics. – 2021. – doi:10.1109/SMC52423.2021.9659271.
7. Flops counting tool for neural networks in pytorch framework (Aug 2024) [Online]. Available: <https://github.com/sovrasov/flops-counter.pytorch>
8. Getty Images (Aug 2024) [Online]. Available: <https://www.gettyimages.com>
9. Girshick R. Region-based convolutional networks for accurate object detection and segmentation / R. Girshick, J. Donahue, T. Darrell, J. Malik // IEEE transactions on pattern analysis and machine intelligence. – 2015. – Vol. 38, № 1.
10. Keras FLOP Estimator (Aug 2024) [Online]. Available: https://github.com/ckyrkou/Keras_FLOP_Estimator
11. Lin T.-Y. Focal loss for dense object detection / T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2018.
12. Liu W. SSD: Single Shot MultiBox Detector / W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, Ch-Y. Fu, A.C. Berg // ECCV. – 2016. – doi:10.48550/arXiv.1512.02325.
13. Mask Dataset (Aug 2024) [Online]. Available: <https://www.kaggle.com/datasets/andrewmvd/face-mask-detection>
14. Object Detection Metrics (Aug 2024) [Online]. Available: https://github.com/yfpeng/object_detection_metrics
15. Pexels (Aug 2024) [Online]. Available: <https://www.pexels.com>
16. Prados-Torreblanca A. Shape Preserving Facial Landmarks with Graph Attention Networks / A. Prados-Torreblanca, J.M. Buenaposada, L. Baumela. – BMVC, 2022. – doi:10.48550/arXiv.2210.07233.
17. Redmon J. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon, S. Divvala, R. Girshick, A. Farhadi // Computer Vision and Pattern Recognition. – 2016. – doi: 10.1109/CVPR.2016.91.
18. Sethi Sh. Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread / Sh. Sethi, M. Kathuria, T. Kaushik // Journal of Biomedical Informatics. – 2021. – Vol. 120. – doi: .10.1016/j.jbi.2021.103848.
19. Tan M. EfficientNet V2: Smaller Models and Faster Training / M. Tan, Q.V. Le. – ICML, 2021. – doi:10.48550/arXiv.2104.00298.
20. Zhou Shangchen Towards Robust Blind Face Restoration with Codebook Lookup Transformer / Shangchen Zhou et al. – 1 Nov 2022. – arXiv:2206.11253v2 [cs.CV].
21. Unsplash (Aug 2024) [Online]. Available: <https://unsplash.com>
22. Viola P. Rapid object detection using a boosted cascade of simple features / P. Viola, M. Jones // Computer Vision and Pattern Recognition. – 2001. – Vol. 1. – P. 1–511. – doi: 10.1109/CVPR.2001.990517.
23. Viola P. Robust real-time face detection / P. Viola, M. Jones // Computer Vision and Pattern Recognition. – 2001. – P. 747–747. – doi: 10.1109/ICCV.2001.937709.
24. YOLOv7Face (Aug 2024) [Online]. Available: <https://github.com/derronqi/yolov7-face>
25. Zou Z. Object Detection in 20 Years: A Survey / Z. Zou, K. Chen, Zh. Shi, Y. Guo, J. Ye. – 2019. doi:10.48550/arXiv.1905.05055.

Стаття: надійшла до редколегії 10.10.2024

доопрацьована 31.10.2024

прийнята до друку 14.11.2024

ON THE CLASSIFICATION OF FACE IMAGES USING COMPUTER VISION AND MACHINE LEARNING

Y. Yurchenko¹, N. Yurchenko², A. Muzychuk¹

¹*Ivan Franko National University of Lviv,
1, Universytetska str., 79000, Lviv, Ukraine*

e-mail: yuliana.yurchenko1673@gmail.com, anatoliy.muzychuk@lnu.edu.ua,

²*Uzhhorod National University,
14, Universytetska str., Uzhhorod, Ukraine*

e-mail: nataliia.yurchenko@uzhnu.edu.ua

This paper addresses the problem of real-time facial image classification based on certain characteristics of faces. In particular, the work focuses on face mask classification task. Since the volume of open-source annotated datasets of images of people wearing masks is relatively small, the paper proposes an approach to generate corresponding images based on facial landmarks. The use of the synthetic data for training the face mask classifier led to an increase in its accuracy.

In light of the advantage of automating a surveillance for face mask compliance in public places, we integrated the YOLOv7Face face detection model with the trained classifier to develop a face mask detector. A new dataset consisting of images of students and teachers from educational institutions was compiled to evaluate the model. The resulting model achieved a mAP of 87.2% on the newly created dataset for detecting face masks. Additionally, we found that applying CodeFormer for face restoration at an intermediate step can decrease the error rate of the mask detector, though it substantially increases inference time.

Key words: computer vision, convolutional neural network, object detection, face detection, facial landmarks, face mask detection, real-time.