МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»

Кваліфікаційна наукова праця
на правах рукопису

ХАБАРЛАК КОСТЯНТИН СЕРГІЙОВИЧ

УДК 004.93

ДИСЕРТАЦІЯ

МЕТОДИ КЛАСИФІКАЦІЇ ТА СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ НА ОСНОВІ ЗМІНЮВАНИХ ЗГОРТКОВИХ МЕРЕЖ

Спеціальність 122 Комп’ютерні науки
Галузь знань 12 Інформаційні технології

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело
_____ К.С. Хабарлак

Науковий керівник Коряшкіна Лариса Сергіївна, кандидат фізико-математичних наук, доцент

Дніпро – 2023

АНОТАЦІЯ

Хабарлак К.С. Методи класифікації та сегментації зображень на основі змінюваних згорткових мереж. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 «Комп'ютерні науки». – Національний технічний університет «Дніпровська політехніка», Дніпро, 2023.

Згорткові нейронні мережі показують високу якість у розв'язанні задач комп'ютерного зору. Суттєва кількість досліджень присвячена розробці нейронних мереж для їх виконання на потужних серверах, однак в ряді випадків їх використання ускладнюється з таких причин: коли інтернет з'єднання є нестабільним або відсутнє взагалі, коли користувач не погоджується передавати приватні дані із свого пристрою, коли загальний об'єм даних надто великий для передачі з усіх пристроїв на сервер тощо. У разі необхідності обробки зображень на мобільному або малопотужному пристрої виникає цілий ряд проблем:

— такі пристрої мають обмежені обчислювальні ресурси, і мережа на них може виконуватись за неприпустимо довгий для цільової задачі час. А отже, архітектури глибоких згорткових нейронних мереж із великою кількістю параметрів, що показують високу якість на серверах, мають зазнати змін для застосування на мобільних пристроях;

— робота від батареї передбачає мінімізацію кількості обчислень. Через це великий інтерес наукової спільноти спрямований на розробку архітектур мобільних нейронних мереж, що враховують характеристики пристроїв на етапі проектування. Проблемою таких нейронних мереж є необхідність остаточного визначення їх конфігурації до початку процедури навчання, що ви-

магає повтору довгої процедури навчання після кожної корекції конфігурації мережі;

— якщо застосунок необхідно встановити на пристрій Інтернету речей, це додає ще одну категорію пристроїв із меншою обчислювальною потужністю і ставить розробника нейронної мережі перед вибором: або навчити одну мережу, яка буде достатньо швидкою для всіх пристроїв, але потенційно матиме невисоку якість виконання; або ж навчати окрему мережу для кожної категорії пристроїв, що, враховуючи довгий час навчання глибоких нейронних мереж, значно збільшить витрати на розробку системи.

Метою роботи є прискорення навчання і виконання згорткових нейронних мереж для задач класифікації та сегментації зображень без втрат (або з якомога меншими втратами) якості розпізнавання за рахунок розробки змінюваних нейронних мереж і методів їх навчання. Під змінюваною нейронною мережею будемо розуміти згорткову мережу із змінною складністю.

Наукова новизна одержаних результатів:

— вперше для задач класифікації та сегментації зображень розроблені змінювані згорткові нейронні мережі та метод їх навчання, які, на відміну від існуючих, дозволяють обирати одну з конфігурацій із різними обчислювальними складностями під час або після навчання. На наборі даних ImageNet розроблена мережа за ефективністю (в сенсі співвідношення якість розпізнавання/час виконання) зайняла п’яте місце серед 17 провідних архітектур мереж, а на CamVid прискорення виконання склало понад 6 % без втрат якості;

— вперше розроблено метод Λ-шаблонів прискорення оптимізаційного мета-навчання, який, на відміну від існуючих, дозволяє за рахунок зміни складності нейронної мережі зменшити кількість обчислень під час навчання, та таким чином пришвидшити адаптацію мережі до нових класів за малою кількістю прикладів на 7,5 % при втратах якості менше 0,4 %.

В першому розділі розглянуто проблеми впровадження нейронних ме-

реж в наступних застосунках: пошук та ідентифікація облич, розпізнавання емоцій, антиспуфінг, ідентифікація стану водія, анімація персонажів, пошук ключових точок. Проведено порівняльний аналіз архітектур нейронних мереж:

— для задач класифікації: AlexNet, VGG, ResNet, MobileNetV2, SENet, MnasNet, MobileNetV3;
— для задач сегментації: U-Net, Hourglass, HRNet, CU-Net.

Для кожної мережі наведено її особливості, оцінено обчислювальну складність (за кількістю операцій множення та додавання) та розраховано кількість параметрів; проаналізовано переваги та недоліки.

За проведеним аналізом застосунків мобільних нейронних мереж зазначено важливість зменшення часу їх виконання, не втрачаючи якості розпізнавання зображень, та зміни конфігурацій нейронних мереж під час їх розгортання на пристроях із різними обчислювальними можливостями. Розглянуто існуючі методи прискорення навчання та виконання нейронних мереж.

Також проаналізовано методи мета-навчання, що дозволяють навчити мережу лише за кількома прикладами на клас. MAML є ключовим методом оптимізаційного мета-навчання та є основою великої кількості подальших підходів. Виявлено, що недоліком таких методів є повільна процедура адаптації мережі до нових класів.

Для розробки змінюваних згорткових нейронних мереж і методів їх навчання обґрунтовано вибір в якості базових:

— мережі MobileNetV2, котра широко використовується для вирішення багатьох практичних проблем комп’ютерного зору, зокрема задачі класифікації;
— мережі U-Net, яка є основою багатьох нейронних мереж, розроблених для задач сегментації;
— мережі CNN4, яка є основою методів оптимізаційного навчання за кіль-

кома прикладами, базовим з яких є метод MAML.

В другому розділі для задач класифікації розроблено змінювану згорткову нейронну мережу, що дозволяє вибирати архітектуру відповідно до обчислювальних можливостей пристроїв. Розроблено метод навчання такої мережі.

Ключовим структурним компонентом змінюваної згорткової мережі (мережі PTA) є згортковий блок PTA, що складається з двох гілок: легкої та важкої. Перша є вдвічі швидшою за другу. Виконувати можна кожну з них окремо або обидві одночасно. Розроблений метод навчання дозволяє обирати конфігурацію такого блоку не лише під час навчання, але й на етапі її виконання.

Роботу розробленої нейронної мережі та методу її навчання перевірено на наборі даних для задачі класифікації ImageNet. Час виконання мережі у порівнянні з оригінальною MobileNetV2 зменшено на 13,74 % при падінні точності (топ 1) на 3,68 %.

Також проведено експерименти на наборі даних антиспуфінгу (задача класифікації) CelebA-Spoof, де мережа PTA перевершила оригінальну за всіма метриками та дозволила зменшити час виконання до 20 %. Зокрема, найкращі отримані метрики (в дужках – результати MobileNetV2): точність 97,85 % (проти 96,74 %), частоти помилок: BPCER 1,98 % (проти 4,18 %), APCER 0,70 % (проти 1,07 %), за ACER 2,13 % (проти 2,63 %). Загальний час навчання PTA моделі зменшено на 14,34 % у порівнянні із MobileNetV2.

В третьому розділі представлено нову мобільну систему контролю доступу із RFID мітками і підсистемою антиспуфінгу, розробленою на основі змінюваних згорткових мереж, яка дозволяє зменшити навантаження на сервер та підвищити захищеність самої системи контролю доступу. Запропонована система контролю доступу включає:

— адміністративну панель для налаштування політик доступу до підприємства;

— систему моніторингу з фільтрами за часом доступу, користувачем та контрольованими дверми з RFID-мітками;
— мобільний додаток, що здійснює пошук облич та здійснює перевірку зображення на спуфінг. Додаток створений для реєстрації і відмикання контрольованих дверей;
— серверну програму, яка оброблює, зберігає та надає дані для додатків на ПК і смартфоні.

Впровадження розробленої системи дозволяє знизити вартість систем контролю доступу за рахунок заміни стаціонарного RFID-сканера на дешеву мітку, а також відмовитися від встановлення камер відеоспостереження, оскільки користувач робить фотографію на свій мобільний телефон, коли відмикає двері, а його фотографія перевіряється системою антиспуфінгу.

В четвертому розділі розроблені блоки PTA інтегровано в мережу U-Net, яка використовується для задачі сегментації зображень. Навчання змінюваної згорткової мережі проведено на наборі даних CamVid. Мережу розгорнуто на крайовому, мобільних, персональних комп'ютерах та графічному процесорі. Показано, що остаточну навчену мережу PTA можна перемикати під час виконання між шістьма конфігураціями, що відрізняються часом виконання та якістю. Важливо, що всі конфігурації мають вищу якість, ніж оригінальна мережа U-Net (із $\text{Dice}_{\text{score}} = 0{,}8583$). За усіма пристроями (в середньому) прискорення виконання мережі склало 6,09 % з $\text{Dice}_{\text{score}} = 0{,}8647$.

В п'ятому розділі описано розроблений метод Λ-шаблонів прискорення оптимізаційного мета-навчання, який, на відміну від існуючих, дозволяє змінювати кількість обчислень у методі зворотного розповсюдження помилки, за рахунок чого зменшено час адаптації мережі до нових класів за малою кількістю прикладів. Експериментально виявлено 2 найкращих шаблони, які дозволили зменшити час адаптації на 7,51 % (падіння точності: 0,33 %) або на 14,96 % (падіння точності: 1,25 %).

Метод Λ-шаблонів продемонстрував підвищення точності класифікації у випадку однокрокового навчання за кількома прикладами. Найбільше покращення отримано в конфігурації по 5 прикладів на 5 класів, де, наприклад, метод MAML за один крок адаптації демонстрував точність 20,4 %, що є показником близьким до випадкового вгадування, а метод Λ-шаблонів – 54,8 %.

Практичне значення одержаних результатів:

— розроблену змінювану згорткову нейронну мережу можна використовувати для розв'язання задач класифікації та сегментації будь-яких зображень, як на серверах, комп'ютерах, так і на портативних, мобільних пристроях;

— розроблений мобільний застосунок, який опрацьовує вхідне відео з камери в реальному часі прямо на мобільному пристрої, гнучко налаштовується для роботи із будь-якими задачами класифікації та сегментації зображень та може бути використаний, зокрема, на транспортних підприємствах для відстеження стану водія під час керування в умовах відсутнього або повільного доступу до мережі Інтернет;

— розроблений застосунок із методом Λ-шаблонів прискорення мета-навчання дозволяє пришвидшити навчання нейронної мережі для задачі класифікації у випадках, коли навчальний набір є малим через складність або коштовність збору такого набору даних, наприклад, в системах відстеження рухів та анімації обличчя;

— розроблену мобільну систему контролю доступу можна використовувати на виробничих підприємствах задля забезпечення безпеки доступу до технологічного обладнання і дверей. За рахунок використання RFID міток та вбудованої підсистеми антиспуфінгу розроблена система є досить дешевою у впровадженні порівняно із аналогами.

Ключові слова: згорткова нейронна мережа, змінювана нейронна мережа, машинне навчання, сегментація зображень, класифікація, розпізнавання

зображень, комп’ютерний зір, мобільні пристрої, антиспуфінг, система контролю доступу, час виконання, навчання за кількома прикладами, мета-навчання.

ABSTRACT

Khabarlak K.S. Image Classification and Segmentation Methods Based on Changeable Convolutional Neural Networks. – Qualifying scientific work, the manuscript.

PhD thesis in specialty 122 Computer Science. – Dnipro University of Technology, Dnipro, 2023.

Convolutional neural networks show high quality in solving computer vision tasks. A significant amount of research is devoted to the development of neural networks, that target inference on powerful servers. However, in a number of cases their use is complicated for the following reasons: when the Internet connection is unstable or absent at all, when the user does not agree to share private data from his device, when the data volume is too large to be transferred from all devices to the server, etc. If it is necessary to process images on a mobile or low-power device, a number of problems arise:

— such devices have limited computing resources, and the network inference might be unacceptably long for the target task. Therefore, deep convolutional neural network architectures with many parameters that show high quality on servers need to be modified for mobile applications;

— inference when running on battery implies that the number of computations should be minimized. Because of this, great interest of the scientific community is devoted to the development of mobile neural network architectures that take into account the mobile device limitations at the design stage. Such architectures require the network configuration to be finalized before the start of the training procedure, as a result long training procedure should be repeated after each network architecture adjustment, which is a problem;

— if the application is expected to be installed on an IoT device, this adds

another category of devices with less computing power and presents the neural network developer with a choice: either to train one network that will be fast enough for all devices, but potentially have poor performance; or to train a separate network for each category of devices, which, given the long training time of deep neural networks, will significantly increase the cost of development of the system.

The purpose of the work is to accelerate convolutional neural network training and inference for the tasks of image classification and segmentation without recognition quality loss (or with as little loss as possible) by developing changeable neural networks and their training methods. By changeable neural network we mean a convolutional network with changeable complexity.

Scientific novelty of the obtained results:

— for the first time, the changeable convolutional neural network and its training method were developed for the tasks of classification and segmentation. In contrast to the existing ones, changeable neural networks enable configuration selection among the ones with different computational complexities during or after training. On the ImageNet dataset the developed neural network in terms of efficiency (in the sense of recognition quality/execution time ratio) took fifth place among the 17 considered state-of-the-art neural network architectures on the ImageNet dataset. On the CamVid dataset the speed up is above 6 % without quality loss;

— for the first time, the Λ-patterns method of optimization meta-learning acceleration was developed, which, unlike existing methods, allows to change neural network complexity during training and, consequently, speed up the neural network few-shot adaptation by 7.5 % with the quality loss below 0.4 %.

In the first chapter the neural network implementation problems are considered in the following applications: face search and recognition, emotion recognition, anti-spoofing, driver state tracking, character animation, facial landmark detection. A comparative analysis of neural network architectures was

carried out for the tasks of:

— classification: AlexNet, VGG, ResNet, MobileNetV2, SENet, MnasNet, MobileNetV3;
— segmentation: U-Net, Hourglass, HRNet, CU-Net.

For each network, its features are described, computational complexity is estimated (by the number of multiply-add operations), and the number of parameters is calculated; advantages and disadvantages are analyzed.

Based on the analysis of mobile neural network applications, importance of neural network inference time reduction without losing image recognition quality, and of ability to change the neural network configuration when it is deployed on devices with different computing capabilities were noted. Existing methods of neural network training and inference were considered.

Also, meta-learning methods were analyzed, that enable few-shot neural network training. MAML is the key optimization meta-learning method, it serves as a base for many further approaches. It was discovered, that disadvantage of such methods is slow neural network adaptation to the new classes.

For the development of changeable convolutional neural networks and their training methods, the selection as basic architectures is substantiated of:

— the MobileNetV2 network, which is widely used to solve many practical problems of computer vision, in particular, classification problems;
— the U-Net network, which is the basis of many neural networks developed for segmentation tasks;
— the CNN4 network, which serves as a foundation for optimization few-shot learning methods with MAML being the base method.

In the second chapter, the changeable convolutional neural network is developed for the classification tasks, which allows choosing the architecture according to the computing capabilities of the devices. A method of training such a network has been developed.

The key structural component of the changeable convolutional network (PTA network) is the PTA convolutional block, which consists of two branches: light and heavy. The first is twice as fast as the second. It is possible to infer each of them exclusively or both at the same time. The developed training method enabled the configuration selection of the block not only during training, but also during inference.

The evaluation of the developed neural network and its training method was performed on the ImageNet image classification dataset. Inference time of the network compared to the original MobileNetV2 is reduced by 13.74 % for the accuracy (top 1) loss of 3.68 %.

Experiments were also performed on the CelebA-Spoof anti-spoofing dataset (classification task), where the PTA network outperformed the original one in all metrics and reduced the inference time by up to 20 %. In particular, the best obtained metrics (MobileNetV2 results are shown in brackets): accuracy 97.85 % (versus 96.74 %), error rates: BPCER 1.98 % (versus 4.18 %) , APCER 0.70 % (versus 1.07 %), for ACER 2.13 % (versus 2.63 %). The total training time of the PTA model is reduced by 14.34 % compared to MobileNetV2.

The third chapter presents a new mobile access control system with RFID tags and a built-in anti-spoofing subsystem developed based on the changeable convolutional network, which allows to reduce the server load and to increase access control system security. The proposed access control system includes:

— administrative panel for configuring enterprise access policies;
— monitoring system with filters by access time, user and controlled doors with RFID tags;
— mobile application that performs face search and anti-spoofing check. The application is created for the controlled door registration and unlocking;
— a server program that processes, stores and provides data for applications on PCs and smartphones.

The implementation of the developed system makes it possible to reduce the cost of access control systems by replacing the stationary RFID scanner with a cheap tag, as well as by avoiding the installation of video surveillance cameras, since the user takes a photo on his mobile phone when he unlocks the door, and his photo is checked by the anti-spoofing system.

In the fourth chapter, the developed PTA blocks are integrated into the U-Net network, which is used for the image segmentation task. The training of the modified convolutional network was performed on the CamVid dataset. The network is deployed on edge, mobile, personal computers and graphical processors. It is shown that the final trained PTA network can be switched at runtime between six configurations differing in inference time and quality. Importantly, all configurations are of higher quality than the original U-Net (with $Dice_{score}$ = 0.8583). Across all devices (on average), the network speedup was 6.09 % with $Dice_{score}$ = 0.8647.

In the fifth chapter, the developed method of Λ-patterns for accelerating optimization meta-learning is described, which, unlike the existing ones, allows changing the number of calculations in the backpropagation method, which allows to decrease few-shot learning adaptation time. The 2 best patterns were experimentally identified, which allowed to reduce few-shot adaptation time by 7.51 % (accuracy drop: 0.33 %) or by 14.96 % (accuracy drop: 1.25 %).

The Λ-patterns method demonstrated improved classification accuracy in the case of few-shot training with a single adaptation step. The greatest improvement was obtained in the 5-shot 5-way configuration, where, for example, the MAML method for single-step adaptation showed an accuracy of 20.4 %, which is a value close to random guessing, while Λ-patterns achieved an accuracy of 54.8 %.

Practical significance of the obtained results:

— the developed changeable convolutional neural network can be used to solve tasks of image classification and segmentation, both on servers, computers,

and on portable, mobile devices;

— the developed mobile application with real-time camera video processing directly on the mobile device, can be flexibly configured to work with any image classification and segmentation task. It can be used, in particular, in transport enterprises for real-time driver status tracking in conditions of no or slow access to the Internet;

— the developed application with Λ-patterns meta-learning acceleration method allows to speed up neural network training for the classification problem in cases where the training set is small due to high cost or complexity of dataset collection, for example, in motion tracking and face reenactment systems;

— the developed mobile access control system can be used in manufacturing enterprises to ensure technological equipment and doors access safety. Thanks to the use of RFID tags and a built-in anti-spoofing subsystem, it is quite cheap to implement the developed system compared to alternatives.

Keywords: Convolutional Neural Network, Changeable Neural Network, Machine Learning, Image Segmentation, Classification, Image Recognition, Computer Vision, Mobile Devices, Anti-Spoofing, Access Control System, Inference Time, Few-Shot Learning, Meta-Learning.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні наукові результати дисертації:

Публікації у фахових виданнях України:

1. *Khabarlak K. S.*, *Koriashkina L. S.* Mobile Access Control System Based on RFID Tags and Facial Information // Bulletin of National Technical University ”KhPI”. Series: System Analysis, Control and Information Technologies. 2020. Т. 2, № 4. С. 69–74. DOI: 10.20998/2079-0023.2020.02.12. URL: http://samit.khpi.edu.ua/article/view/2079-0023.2020.02.12 **(Фаховий (категорія Б))**.
2. *Хабарлак К. С.* Особливості роботи методів пошуку облич на мобільних пристроях // System Technologies. 2021. Т. 6, № 137. С. 34–45. DOI: 10.34185/1562-9945-6-137-2021-04. URL: https://journals.nmetau.edu.ua/index.php/st/issue/view/118/97 **(Фаховий (категорія Б))**.
3. *Khabarlak K.* Post-Train Adaptive U-Net for Image Segmentation // Information Technology: Computer Science, Software Engineering and Cyber Security. 2022. № 2. С. 73–78. DOI: 10.32782/IT/2022-2-8. URL: https://doi.org/10.32782/IT/2022-2-8 **(Фаховий (категорія Б))**.

Публікації у виданнях, які проіндексовані у міжнародних наукометричних базах Web of Science та Scopus:

1. *Khabarlak K.*, *Koriashkina L.* Fast Facial Landmark Detection and Applications: A Survey // Journal of Computer Science and Technology. 2022. Квіт. Т. 22, № 1. С. 12–41. DOI: 10.24215/16666038.22.e02.

URL: https://journal.info.unlp.edu.ar/JCST/article/view/1972 **(Scopus, Web of Science, закордонний журнал)**.

2. *Khabarlak K. S.* Faster Optimization-Based Meta-Learning Adaptation Phase // Radio Electronics, Computer Science, Control. 2022. Квіт. № 1. С. 82–92. DOI: 10.15588/1607-3274-2022-1-10. URL: http://ric.zntu.edu.ua/article/view/254615 **(Web of Science, Фаховий (категорія А))**.
3. *Khabarlak K. S.*, *Koriashkina L. S.* Scoping Adversarial Attack for Improving Its Quality // Radio Electronics, Computer Science, Control. 2019. Трав. № 2. С. 108–118. DOI: 10.15588/1607-3274-2019-2-12. URL: http://ric.zntu.edu.ua/article/view/178284 **(Web of Science, Фаховий (категорія А))**.

Наукові праці, які засвідчують апробацію матеріалів дисертації:

Авторське свідоцтво:

1. Комп'ютерна програма «Мобільна нейромережева система пошуку облич із анті-спуфінгом» : авт. свід. України №110917 / К. С. Хабарлак. 11.01.2022

Матеріали конференцій та тези доповідей:

1. *Khabarlak K.* Post-Train Adaptive MobileNet for Fast Anti-Spoofing // Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security, Khmelnytskyi, Ukraine, March 23–25. Т. 3156. CEUR-WS.org, 2022. С. 44–53. (CEUR Workshop Proceedings). URL: http://ceur-ws.org/Vol-3156/keynote5.pdf **(Scopus)**
2. *Хабарлак К. С.*, *Коряшкіна Л. С.* Деякі особливості гіперпараметрів глибоких нейронних мереж // III Всеукраїнська Інтернет-конференція здо-

бувачів вищої освіти і молодих учених «Інформаційні технології: теорія і практика». Харків, 03.2020. С. 98–99

3. *Khabarlak K.*, *Koriashkina L.* Top image classification accuracy through hyperparameter search // 15th International Forum for Students and Young Researchers “Widening our horizons”. Dnipro, 05.2020. С. 271–274
4. *Khabarlak K.* Mobile Application for RFID Access Control System // V міжнародна науково-практична конференція «Прикладні науково-технічні дослідження». Івано-Франківськ, 04.2021. С. 99–100
5. *Хабарлак К. С.* Анти-спуфінг для системи контролю доступу із RFID мітками // Збірник матеріалів III Всеукраїнської конференції «Теоретико-практичні проблеми використання математичних методів і комп’ютерно-орієнтованих технологій в освіті та науці». Київ, 04.2021. С. 141–142
6. *Хабарлак К. С.* On Face Detection and Anti-Spoofing in Mobile Access Control // Тези доповідей VIII Міжнародної науково-практичної конференції «Інформаційні технології в освіті, науці і виробництві (ІТОНВ-2021)». Луцьк, 05.2021. С. 181–183
7. *Хабарлак К. С.*, *Коряшкіна Л. С.* Тестування швидкості виконання алгоритмів пошуку обличчь для системи контролю доступу // VII Всеукраїнська науково-технічна конференція молодих учених, аспірантів та студентів «Автоматизація, контроль та управління: пошук ідей та рішень» (АКУ-2021). Покровськ, 05.2021. С. 60–61
8. *Хабарлак К. С.* Про адаптацію мета-навчання нейронних мереж // Матеріали міжнародної конференції II International Scientific Symposium “Intelligent Solutions-S”. Ужгород, 09.2021. С. 81
9. *Хабарлак К. С.* Адаптація мета-навчання на частковому шаблоні // Матеріали VII міжнародної науково-технічної конференції Комп’ютерне моделювання та оптимізація складних систем. Дніпро, 11.2021.

С. 121–122

10. *Хабарлак К. С.* Аналіз шаблонів адаптації мета-навчання // Матеріали IX Всеукраїнської науково-технічної конференції студентів, аспірантів та молодих вчених «Молодь: наука та інновації». Дніпро, 11.2021. С. 328–329
11. *Хабарлак К. С.* Прискорене навчання нейронної мережі за декількома прикладами // XVI Міжнародна конференція з проблем використання інформаційних технологій в освіті, науці та промисловості. Дніпро, 12.2021. С. 62–64
12. *Хабарлак К. С.* Зміна архітектури нейронної мережі після навчання // Тези дев'ятої міжнародної науково-технічної конференції Інформатика, управління та штучний інтелект. Харків – Краматорськ, 05.2022. С. 135
13. *Хабарлак К. С.* Нейромережний пошук об'єктів на мобільних пристроях // Матеріали XII Всеукраїнської науково-технічної конференції студентів, аспірантів та молодих вчених «Наукова весна». Дніпро, 05.2022. С. 171
14. *Хабарлак К. С.* Адаптивна після навчання нейронна мережа // Тези V Всеукраїнської Інтернет-конференції здобувачів вищої освіти і молодих учених Інформаційні технології: теорія і практика. Запоріжжя, 06.2022. С. 20–21
15. *Хабарлак К. С.* Нейро-мережева система класифікації із конфігурацією після навчання // Матеріали X Міжнародної науково-технічної конференції студентів, аспірантів і молодих вчених «Молодь: наука та інновації». Дніпро, 11.2022. С. 383
16. *Khabarlak K.* Semantic segmentation with Post-Train Adaptive Neural Network // Тези XI міжнародної науково-практичної конференції «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій». Запоріжжя, 12.2022. С. 124–125

17. *Хабарлак К. С.* Конфігурація після навчання нейронної мережі для сегментації зображень // Матеріали XIII Міжнародної науково-технічної конференції аспірантів та молодих вчених «Наукова весна». Дніпро, 03.2023. С. 194–195
18. *Хабарлак К. С.* Проблеми нейронних мереж для розпізнавання на пристроях із різними обчислювальними можливостями // Тези VI Всеукраїнської науково-практичної Інтернет-конференції здобувачів вищої освіти і молодих учених Інформаційні технології: теорія і практика. Харків, 03.2023. С. 101–102

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

Змінювана згорткова мережа PTA

ACER	Середня частота помилок класифікації (Average Classification Error Rate)
APCER	Частота помилок класифікації представлення атак (Attack Presentation Classification Error Rate)
BPCER	Частота помилок класифікації достовірного представлення (Bona Fide Presentation Classification Error Rate)
GUID	Глобально унікальний ідентифікатор (Globally Unique Identifier)
NFC	Комунікація ближнього поля (Near-Field Communication)
PTA	Адаптивний після навчання (Post-Train Adaptive)
PTA-BBB	Конфігурація PTA: [Обидві, Обидві, Обидві] гілки
PTA-HHH	Конфігурація PTA: [Важка, Важка, Важка] гілки
PTA-HHL	Конфігурація PTA: [Важка, Важка, Легка] гілки
PTA-HLH	Конфігурація PTA: [Важка, Легка, Важка] гілки
PTA-LHH	Конфігурація PTA: [Легка, Важка, Важка] гілки
PTA-LLL	Конфігурація PTA: [Легка, Легка, Легка] гілки
RFID	Радіочастотна ідентифікація (Radio Frequency Identification)

Метод Λ-шаблонів прискорення оптимізаційного мета-навчання

Λ	Лямбда шаблон
$\mathcal{L}$	Функція втрат
$\Phi(\theta, \cdot)$	Нейронна мережа із вагами θ
$\rho(\mathcal{T})$	Простір задач
$\theta_l^{(i,j)}$	Матриця параметрів l-го шару мережі, отриманих для i-ї задачі на j-й ітерації адаптації
H	Кількість задач, що використовуються на одній ітерації навчання
K-прикл.	Конфігурація мета-навчання за кількома прикладами, де для кожного класу надається по K навчальних прикладів (K-shot)
N-клас.	Конфігурація мета-навчання за кількома прикладами між N класами (N-way)
P	Кількість кроків адаптації
$Q = \{X_Q, y_Q\}$	Набір запитів
$S = \{X_S, y_S\}$	Набір підтримки
X	Дані, що подаються на вхід до мережі
y	Унітарний вектор, що відповідає вірному номеру класа зображення
MAML	Метод мета-навчання довільної моделі (Model-Agnostic Meta-Learning)

ВСТУП

Актуальність роботи. Нейронні мережі показують високу якість у розв'язанні задач, специфічних для людини. Початкові практичні застосування нейронних мереж фокусувалися на потужних серверах із декількома графічними процесорами та стабільним підключенням до електромережі. Однак, використання клієнт-серверних програм на крайових та мобільних пристроях пов'язано із суттєвими складностями, наприклад, коли інтернет з'єднання є нестабільним або відсутнє взагалі, коли користувач не погоджується передавати приватні дані із свого пристрою, або ж коли загальний об'єм даних надто великий, щоб з усіх пристроїв передати дані на сервер.

Одним із ключових типів даних, що генеруються користувачем та потребують миттєвої обробки й аналізу, є зображення. В багатьох випадках бажаним для аналізу є використання згорткових нейронних мереж, а за перелічених вище причин обробку необхідно проводити на мобільному пристрої користувача. Однак, задача розробки, навчання і виконання нейронних мереж, зокрема згорткових, на мобільних пристроях має цілий ряд проблем:

— мережа може не поміститися або виконуватися за неприпустимо довгий час на мобільному пристрої, оскільки такі пристрої мають значно обмежену постійну та оперативну пам'ять, обчислювальні ресурси. А, отже, для розгортання глибоких згорткових нейронних мереж із великою кількістю параметрів, що показують високу якість на серверах, їх архітектури мають бути скореговані;

— слід враховувати необхідність роботи мобільного пристрою від батареї та, відповідно, мінімізацію кількості обчислень. Значна увага наукової спільноти приділяється розробці архітектур мобільних нейронних ме-

реж, що враховують обмеження мобільних пристроїв на етапі проектування. Проблемою таких нейронних мереж є необхідність остаточного визначення конфігурації нейронної мережі до початку процедури навчання;

— якщо застосунок необхідно встановити на пристрій Інтернету речей, це додає ще одну категорію пристроїв із меншою обчислювальною потужністю і ставить розробника нейронної мережі перед вибором: або навчити одну мережу, яка буде достатньо швидкою для всіх пристроїв, але потенційно матиме невисоку якість виконання; або ж навчати окрему мережу для кожної категорії пристроїв, що, враховуючи довгий час навчання глибоких нейронних мереж, значно збільшить витрати на розробку системи. Обидва підходи не є оптимальними.

Базою досліджень стали роботи науковців: K. He, M. Sandler, A. Howard (в області побудови архітектур нейронних мереж та методів їх навчання), C. Finn, A. Antoniou, Z. Li (в області побудови методів мета-навчання).

Сучасні дослідження спрямовані насамперед на знаходження нових підходів до адаптації архітектур нейронних мереж до мобільних пристроїв за допомогою розробки нових ефективних блоків для нейронних мереж, модифікації процедури навчання та розробки динамічних нейронних мереж, що додатково змінюються вже після завершення навчання. Однак, механізм адаптації нейронної мережі вимагає додаткові обчислювальні ресурси пристрою, тому відношення якості до часу виконання, зазвичай, є невисоким.

Відтак, розробка підходів щодо зменшення часу виконання згорткових нейронних мереж із високою якістю розпізнавання з можливістю вибору конфігурації мережі після навчання відповідно до обмежених обчислювальних ресурсів пристроїв, на яких вона розгортається, є актуальним напрямком наукових досліджень.

Зв’язок роботи з науковими програмами, планами, темами. Дисер-

таційна робота виконувалася у Національному технічному університеті «Дніпровська політехніка» згідно до плану науково-дослідних робіт кафедри Системного аналізу та управління, в рамках науково-дослідних робіт: «Розробка нових мобільних інформаційних технологій для ідентифікації особи та класифікації об'єктів навколишнього світу», номер державної реєстрації 0121U109787; «Розробка нових адаптивних інформаційних технологій для розпізнавання об'єктів навколишнього світу», номер державної реєстрації 0123U100012.

Мета і завдання дослідження. *Метою роботи* є прискорення навчання і виконання згорткових нейронних мереж для задач класифікації та сегментації зображень без втрат (або з якомога меншими втратами) якості розпізнавання за рахунок розробки змінюваних нейронних мереж і методів їх навчання. Під змінюваною нейронною мережею будемо розуміти згорткову мережу із змінною складністю.

Для досягнення цієї мети необхідно вирішити такі *основні задачі*:

1. Провести аналіз сучасних архітектур згорткових нейронних мереж, що використовуються для розв'язання задач комп'ютерного зору. Виокремити переваги та недоліки таких мереж з огляду на практичні застосунки, зокрема мобільні. Визначити архітектури для подальших досліджень.
2. Розробити змінювану згорткову нейронну мережу для задачі класифікації за рахунок вбудови в базову додаткового згорткового блоку, що дозволить обирати одну з конфігурацій мережі із різними обчислювальними складностями протягом навчання або під час виконання. Розробити метод навчання отриманої згорткової нейронної мережі. Експериментально дослідити час, якість розробленої змінюваної мережі на визнаному наборі даних класифікації ImageNet, провести порівняння з іншими існуючими нейронними мережами. Здійснити програмну ре-

алізацію системи антиспуфінгу, як однієї з задач класифікації, провести заміри часу виконання на пристроях із різними обчислювальними можливостями та підтвердити прискорення виконання мережі, оцінити якість мережі за метриками антиспуфінгу на наборі даних CelebA-Spoof.

3. Впровадити змінювану згорткову мережу в підсистему антиспуфінгу мобільної системи контролю доступу із RFID мітками, де аналіз зображень виконувався би на смартфоні користувача із метою зменшення вартості впровадження такої системи та підвищення її безпеки. Розробити застосунки для комп'ютерів, смартфонів.
4. Застосувати розроблену змінювану згорткову нейронну мережу та метод її навчання для задачі сегментації зображень. Провести експерименти та оцінити якість мережі на наборі даних CamVid, підтвердити прискорення роботи мережі за рахунок вибору її конфігурації після завершення навчання.
5. Розробити метод Λ-шаблонів для прискорення навчання методів оптимізаційного мета-навчання. Експериментально підтвердити прискорення навчання на наборі даних CIFAR-FS за допомогою запропонованого методу за мінімальних втрат якості.

Об'єктом дослідження є процеси навчання та виконання згорткових нейронних мереж, а також їх використання на малопотужних пристроях в задачах комп'ютерного зору.

Предметом дослідження є архітектури згортових нейронних мереж та методи їх навчання.

Методи дослідження. Для проведення досліджень використовуються методи машинного навчання, принципи побудови згорткових нейронних мереж, методи мета-навчання, методи градієнтного спуску та зворотного розповсюдження помилки для навчання нейронної мережі, теорія алгоритмів та прин-

ципи функціонального й об’єктно-орієнтовного програмування для побудови програмного забезпечення.

Наукова новизна одержаних результатів полягає в наступному:

Вперше:

— для задач класифікації та сегментації зображень розроблені змінювані згорткові нейронні мережі та метод їх навчання, які, на відміну від існуючих, дозволяють обирати одну з конфігурацій із різними обчислювальними складностями під час або після навчання. На наборі даних ImageNet розроблена мережа за ефективністю (в сенсі співвідношення якість розпізнавання/час виконання) зайняла п’яте місце серед 17 провідних архітектур мереж, а на CamVid прискорення виконання склало понад 6 % без втрат якості;

— розроблено метод Λ-шаблонів прискорення оптимізаційного мета-навчання, який, на відміну від існуючих, дозволяє за рахунок зміни складності нейронної мережі зменшити кількість обчислень під час навчання, та таким чином пришвидшити адаптацію мережі до нових класів за малою кількістю прикладів на 7,5 % при втратах якості менше 0,4 %.

Отримали подальший розвиток:

— вдосконалено систему контролю доступу із RFID мітками, в якій, на відміну від інших, підсистему антиспуфінгу розгорнуто на мобільному пристрої, що дозволило за рахунок використання змінюваних згорткових мереж зменшити навантаження на пристрій до 20 % та підвищити захищеність самої системи контролю доступу;

— нейронні мережі MobileNetV2 та U-Net за рахунок впровадження додаткових згорткових блоків, які дозволили змінювати конфігурацію мережі в процесі та після навчання з урахуванням потужності цільового пристрою. У порівнянні з оригінальними мережами для задачі класифікації прискорення досягає 20 %, для задачі сегментації 6 %;

— метод оптимізаційного мета-навчання Model-Agnostic Meta-Learning

за рахунок впровадження Λ-шаблонів, що дозволило підвищити ефективність процедури адаптації до нових класів за малою кількістю навчальних прикладів від 7 % до 13 % в залежності від конфігурації Λ-шаблону.

Практичне значення одержаних результатів:

— розроблену змінювану згорткову нейронну мережу можна використовувати для розв'язання задач класифікації та сегментації будь-яких зображень, як на серверах, комп'ютерах, так і на портативних, мобільних пристроях;

— розроблений мобільний застосунок, який опрацьовує вхідне відео з камери в реальному часі прямо на мобільному пристрої, гнучко налаштовується для роботи із будь-якими задачами класифікації та сегментації зображень та може бути використаний, зокрема на транспортних підприємствах для відстеження стану водія під час керування в умовах відсутнього або повільного доступу до мережі Інтернет;

— розроблений застосунок із методом Λ-шаблонів прискорення мета-навчання дозволяє пришвидшити навчання нейронної мережі для задачі класифікації у випадках, коли навчальний набір є малим через складність або коштовність збору такого набору даних, наприклад, в системах відстеження рухів та анімації обличчя;

— розроблену мобільну систему контролю доступу можна використовувати на виробничих підприємствах задля забезпечення безпеки доступу до технологічного обладнання і дверей. За рахунок використання RFID міток та вбудованої підсистеми антиспуфінгу розроблена система є досить дешевою у впровадженні порівняно із аналогами.

Впровадження одержаних результатів Впровадження одержаних результатів виконано в Придніпровському науковому центрі для розробки системи безпеки дорожнього руху, що включає контроль стану водіїв під час керування транспортними засобами; в Інституті геотехнічної механіки ім.

М. С. Полякова, а саме на виробничих підприємствах гірничо-металургійної галузі в системах контролю доступу до технологічного обладнання і дверей; в НТУ «Дніпровська політехніка» в навчальному процесі на кафедрі системного аналізу та управління при викладанні дисципліни «Самонавчання складних систем» для підготовки магістрів спеціальності 124 «Системний аналіз» та дисципліни «Methods and systems of artificial intelligence» для підготовки бакалаврів (іноземних студентів) спеціальності 122 «Комп'ютерні науки».

Особистий внесок здобувача. Всі основні результати дисертаційної роботи отримані автором особисто. Одноосібно опубліковані праці [1–19]. У друкованих працях, опублікованих у співавторстві здобувачеві належать:

— [20] – пошук релевантної літератури, підготовка схем архітектур нейронних мереж, аналіз методів пошуку ключових точок обличчя, їх якості та узагальнення ідей, аналіз їх сильних та слабких сторін в залежності від ряду факторів, підбиття підсумків та висвітлення можливих подальших напрямків роботи, аналіз практичних застосунків;

— [21] – розробка повнозв'язної та згорткової нейронних мереж, аналіз їх поведінки та розробка методу змагальної атаки;

— [22] – ідея системи контролю доступу на основі RFID міток, проектування та розробка програмного забезпечення для персонального комп'ютера та мобільного телефону;

— [23; 24] – аналіз залежності якості згорткової нейронної мережі та часу навчання від архітектури мережі та її гіперпараметрів;

— [25] – розробка системи діагностики швидкості виконання згорткових нейронних мереж для мобільного застосунку.

Апробація результатів дисертації. Результати та основні положення роботи доповідалися та обговорювалися на:

— 3rd International Workshop on Intelligent Information Technologies

& Systems of Information Security (CEUR Workshop Proceedings), Khmelnytskyi, Ukraine, March 23–25, 2022 (Scopus);

— III, V, VI Всеукраїнській Інтернет-конференції здобувачів вищої освіти і молодих учених «Інформаційні технології: теорія і практика», (Харків, 2020), (Запоріжжя, 2022), (Харків, 2023);
— XV International Forum for Students and Young Researchers “Widening our horizons”, Dnipro, 2020;
— V міжнародна науково-практична конференція «Прикладні науково-технічні дослідження», Івано-Франківськ, 2021;
— III Всеукраїнській конференції «Теоретико-практичні проблеми використання математичних методів і комп’ютерно-орієнтованих технологій в освіті та науці», Київ, 2021;
— VIII Міжнародній науково-практичній конференції «Інформаційні технології в освіті, науці і виробництві (ІТОНВ-2021)», Луцьк, 2021;
— VII Всеукраїнській науково-технічній конференції молодих учених, аспірантів та студентів «Автоматизація, контроль та управління: пошук ідей та рішень» (АКУ-2021), Покровськ, 2021;
— II міжнародній конференції International Scientific Symposium “Intelligent Solutions-S”, Ужгород, 2021;
— VII міжнародній науково-технічній конференції «Комп’ютерне моделювання та оптимізація складних систем», Дніпро, 2021;
— IX Всеукраїнській та X Міжнародній науково-технічній конференції студентів, аспірантів та молодих вчених «Молодь: наука та інновації», (Дніпро, 2021), (Дніпро, 2022);
— XVI Міжнародній конференції з проблем використання інформаційних технологій в освіті, науці та промисловості, Дніпро, 2021;
— IX міжнародній науково-технічній конференції «Інформатика, управління та штучний інтелект», Харків – Краматорськ, 2022;

— XII Всеукраїнській, XIII міжнародній науково-технічній конференції студентів, аспірантів та молодих вчених «Наукова весна», Дніпро, 2022, 2023;
— XI міжнародній науково-практичній конференції «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій». Запоріжжя, 2022.

Публікації. За матеріалами дисертації опубліковано 25 робіт, з яких 3 статті включено до міжнародних наукометричних баз Scopus та Web of Science, 3 статті включено до переліку фахових видань (категорія Б), затверджених МОН України за спеціальністю дисертації, 1 авторське право на твір та 18 – публікації у матеріалах конференцій (у тому числі 11 міжнародних).

Структура та обсяг дисертації. Дисертація складається із анотації, вступу, 5 розділів, висновків, списку використаних джерел та 4 додатків. Робота містить 169 сторінок, у тому числі: 159 сторінок основного тексту, 22 рисунки, 25 таблиць, список використаних джерел налічує 189 найменувань.

РОЗДІЛ 1
АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ. ПРОБЛЕМИ СИСТЕМ КЛАСИФІКАЦІЇ ТА СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ ДЛЯ ПРИСТРОЇВ ІЗ РІЗНИМИ ОБЧИСЛЮВАЛЬНИМИ МОЖЛИВОСТЯМИ

Дослідження цього розділу спрямовані на аналіз методів розв’язання задач класифікації, пошуку ключових точок та сегментації на основі нейронних мереж; виокремлення переваг та недоліків розглянутих підходів в контексті їх впровадження в застосунки. Для цього розв’язано наступні задачі:

1. Проаналізовано сучасні архітектури глибоких згорткових нейронних мереж. Обґрунтовано вибір базових мереж для реалізації змінюваної мережі для задач класифікації та сегментації.
2. Показано важливість використання нейронних мереж для розв’язання задач комп’ютерного зору, в тому числі із обчисленнями прямо на мобільному пристрої. Розглянуто та проаналізовано застосунки, практичні підходи до покращення якості, використані архітектури; виокремлено проблеми їх практичного впровадження. Показано важливість розробки архітектур нейронних мереж, що фокусуються не лише на якості розпізнавання зображень, але й на часі виконання.
3. Проведено узагальнення існуючих модифікацій нейронних мереж на прикладі задачі пошуку ключових точок, як такої, що можна розв’язати або прямим пошуком координат, або через побудову теплових карт, що дозволяє розглянути на одній проблемі архітектури мереж різних типів.
4. Проаналізовано підходи щодо навчання згорткових нейронних мереж за кількома прикладами, зазначено переваги та недоліки підходів.

1.1 Розв'язання задачі класифікації зображень за допомогою згорткових нейронних мереж

Розробка точних методів комп'ютерного зору допомагає вирішувати складні проблеми класифікації зображень, виявлення об'єктів і семантичної сегментації в багатьох задачах, зокрема для автономного водіння [26], обробки супутникових [27] та медичних зображень [28], виявлення та розпізнавання облич, розпізнавання емоцій, визначення ключових точок, відстеження стану водія [20], антиспуфінгу [1].

Початкові спроби розв'язати проблему класифікації зображень базувалися на вилученні ознак об'єктів за допомогою вручну створених функцій (наприклад, фільтрів Хаара [29], дескрипторів HOG [30] тощо) із подальшим додаванням деякого класифікатора (наприклад, градієнтного бустінгу, випадкового лісу). Однак у 2012 році у щорічному змаганні ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [31] було показано, що побудова згорткової нейронної мережі та її наскрізне навчання дає набагато кращі результати. Нейронна мережа-переможець отримала назву **AlexNet** [32]. Після цього фокусом наступних наукових розробок стали саме нейронні мережі, оскільки вони показують високу якість у вирішенні завдань, властивих людині, таких як класифікація, сегментація зображень або виявлення об'єктів.

В першу чергу нейронні мережі були розроблені для роботи на серверах з графічними процесорами і стабільним джерелом живлення. Однак, використання клієнт-серверних програм на крайових та мобільних пристроях пов'язано із суттєвими складностями, наприклад, у випадках:

— поганого підключення до Інтернету;
— необхідності обробки даних з якомога меншою затримкою;
— коли кількість згенерованих необроблених даних надто велика для надсилання на сервер;

— нарешті, коли дані не можуть залишити пристрій користувача з міркувань безпеки.

У багатьох із цих випадків використання нейронних мереж є бажаним, а обробку слід виконувати безпосередньо на мобільному пристрої, тому дослідження підходів щодо прискорення виконання нейронних мереж, у тому числі на мобільних пристроях, стало одним із найважливіших напрямів досліджень [33–38].

Початкові дослідження глибоких згорткових мереж було зосереджено на пошуку точних конфігурацій для блоків згортки (включаючи розмір ядра та крок), типу операції субдискретизації та функцій активації. Кожен блок цих мереж був простим за структурою та містив одну гілку, як, наприклад, мережа **VGG** [39] з глибиною до 19 шарів. Було помічено, що загалом глибокі нейронні мережі мають кращу якість та здатність до узагальнення, ніж неглибокі. Однак виявилося, що побудова глибших за VGG мереж стикається з проблемою надзвичайно повільного навчання. У [40] було проведено експеримент із навчанням мереж, що складаються з простих згорткових блоків, але мають різну глибину. Перша мережа містила 20 шарів. Другу мережу було побудовано поверх першої шляхом приєднання додаткових шарів, загалом – 56 шарів. Очікувалося, що навіть за відсутності додаткових переваг за рахунок доданих шарів, останні можуть вивчити тотожне відображення, а отже, глибша мережа теоретично мала б точність не меншу, ніж неглибока. Проте збіжність навчання мережі з 56 шарами дуже сповільнилася, що призвело до низької кінцевої точності. Подальше дослідження показало, що градієнти, обчислені в методі зворотного поширення помилки, були занадто малими. Така поведінка відома як проблема зникаючого градієнта.

Щоб протистояти проблемі зникаючого градієнта, автори мережі **ResNet** [40] запропонували доповнити групу блоків тотожнім зв’язком. Таке з’єднання було названо «пропусковим» (skip) або «залишковим» (residual).

Основним блоком мережі ResNet є Bottleneck («вузьке місце») блок, який є групою з 3 згорток із розмірами ядра 1×1, 3×3, 1×1 відповідно. Пропускове з'єднання використовується навколо залишкового блоку так, що вхідні дані першої згортки додаються до результату всього блоку. Крім того, щоб обмежити кількість необхідних обчислень в блоці, перша 1×1 згортка зменшує кількість каналів у карті ознак, так що важча 3×3 згортка обробляє дані меншого розміру. Нарешті, остання 1×1 згортка повертає (відновлює) кількість каналів. Звідси й назва «вузьке місце». Також в мережі використовується «залишковий» блок, що являє собою дві згортки 3×3 із пропусковим зв'язком навколо блоку.

Автори архітектури **MobileNetV2** [34] вдосконалюють ідеї, запропоновані раніше в мережі ResNet. MobileNetV2 є однією з найбільш широко використовуваних мереж на мобільних пристроях, в ній представлено інвертований залишковий блок, який, на відміну від Bottleneck блоку, збільшує кількість каналів карти ознак всередині блоку, зберігаючи при цьому менше каналів на входах та виходах. Інвертований залишковий блок мережі MobileNetV2 працює наступним чином:

1. Починає обробку вхідних даних за допомогою 1×1 згортки, яка збільшує кількість каналів. Ступінь збільшення контролюється відповідним коефіцієнтом. За згорткою йде пакетна нормалізація [41] та шар активації ReLU6.
2. Далі застосовано 3×3 згортку, що розділяється (Depthwise Separable Convolution), а потім знову пакетну нормалізацію та ReLU6. Згортка, що розділяється, є легкою та використовується для зменшення кількості обчислень. Згортки такого типу не було в оригінальному дизайні ResNet.
3. Нарешті, застосовано згортку 1×1 з пакетною нормалізацією, щоб зменшити кількість каналів до початкової. Пропускове з'єднання ви-

користовується навколо блоку.

Як показано авторами, запропонована архітектура блоку асимптотично зменшує кількість операцій множення-додавання в мережі порівняно з Bottleneck блоком. Мережа MobileNetV2 має параметр ширини, який регулює обчислювальну складність мережі, але його не можна змінювати після завершення навчання.

Наступні роботи розвивались в таких напрямках: у мережі Squeeze-and-Excitation (**SENet**) [42] із метою покращення якості розпізнавання зображень застосовано механізм уваги, який використовується для вибіркового контролю інформації, що проходить через мережу так, аби лише найважливіші компоненти сигналу враховувались в подальших шарах. У **MnasNet** [36] було запропоновано підхід до автоматизованого пошуку нейронної архітектури для мобільних та вбудованих пристроїв. Під час процесу вибору архітектури нейронної мережі MnasNet, найкращу мережу було обрано на основі замірів швидкості виконання на реальному мобільному пристрої. **MobileNetV3** [35] також покращила попередні підходи завдяки використанню інших функцій активації та методу пошуку архітектури мережі. Було введено велику та малу конфігурації.

Динамічні нейронні мережі є перспективним напрямком досліджень [43–47]. Ідея полягає в тому, щоб дозволити змінювати ваги нейронної мережі або її архітектуру в залежності від необхідного часу виконання або складності вхідних даних. Підходи включають: створення нейронної мережі, яка генерує параметри іншої мережі (HyperNetworks [47]), використання механізму уваги (SENet [42]), часткове виконання мережі ResNet (SACT [45]), прогнозування ваг для згортки сумішшю експертів (CondConv [46]). Однак, суттєвим недоліком таких підходів є складність та збільшенням обчислювальних витрат для реалізації додаткової адаптивності, тобто час виконання може бути в середньому більшим, ніж у звичайної

статичної нейронної мережі.

Також прискорення нейронних мереж можливе шляхом:

— навчання або виконання нейронних мереж із зменшеною точністю операцій з плаваючою комою [48; 49]. Зберігання даних та виконання арифметичних операцій над такими типами даних вимагає апаратної підтримки пристрою на якому працює нейронна мережа;

— квантування, тобто переходу від обчислень із плаваючою комою до цілочисельних [50; 51], які можуть виконуватись швидше за операції з плаваючою комою;

— обрізки нейронної мережі [52; 53], що дозволяє відкинути малозначущі зв'язки або шари нейронної мережі. Результат обрізки є статичним, тобто мережу неможливо перемикати між декількома конфігураціями із різними обчислювальними складностями.

Однією із задач класифікації є антиспуфінг обличчя, мета якого полягає у перевірці чи є обличчя перед камерою справжнім обличчям людини, яка стоїть, чи фотографією, яку тримають, щоб обдурити систему. Антиспуфінг може виконуватися на основі сигналу RGB від звичайної камери, інфрачервоного випромінювання або інформації про глибину від спеціального обладнання. Наприклад, набір даних CASIA-SURF [54] містить відеоінформацію про всі три модальності. Спільне використання трьох модальностей покращує загальну якість, але дані про глибину або інфрачервоні дані зазвичай недоступні та потребують спеціального обладнання. Тому дисертаційне дослідження зосереджено на методах, які використовують лише сигнал RGB, за допомогою якого антиспуфінг все ще можна виконати шляхом пошуку спотворень кольору та форми. Нейронні мережі демонструють високу якість у вирішенні проблеми антиспуфінгу [55; 56], в тому числі у випадку, коли присутній лише сигнал RGB.

Підходи антиспуфінгу на основі нейронних мереж зазвичай базуються на

вже відомих архітектурах нейронних мереж, але із певними вдосконаленнями. У [55] було запропоновано замінити операцію згортки на згортку центральної різниці (Central Difference Convolution), яка краще вловлює градієнти кольорів для покращення ефективності антиспуфінгу. У [56] було представлено мережу AENet із архітектурою ResNet в якості основи. Автори використовують велику кількість анотацій набору даних CelebA-Spoof (представленого в тій самій роботі) для покращення навчання мережі. Одна багатозадачна функція втрат формується з використанням інформації про атрибути обличчя (наприклад, посмішка, сонцезахисні окуляри тощо), умов освітлення фотографії, а також інформації про глибину та віддзеркалення. Відсутню в наборі даних інформацію про глибину та віддзеркалення автори пропонують вивести із зображення RGB за допомогою допоміжної нейронної мережі. Додаткова інформація використовується лише під час навчання і не потрібна при роботі мережі.

1.2 Розв'язання задачі сегментації зображень за допомогою згорткових нейронних мереж

Вищеописані архітектури є універсальними. Однак застосунки виявлення ключових точок та сегментації зображень можуть виграти від архітектур, які розглядають вхідне зображення в кількох роздільних здатностях. Це можна зробити за допомогою мережі піраміди ознак (Feature Pyramid Network, **FPN**) [57], **U-Net**-подібної архітектури [28] або обміну ознаками між кількома масштабами карт ознак (**HRNet**) [58]. В подальшому було показано, що FPN значно менш точний; тому в цій роботі докладно обговорюються лише архітектури на основі U-Net і HRNet.

Архітектуру **Hourglass** [59] спочатку було розроблено для оцінки пози людини. Мережа приймає зображення розміру 256×256 на вхід. Ілюстра-

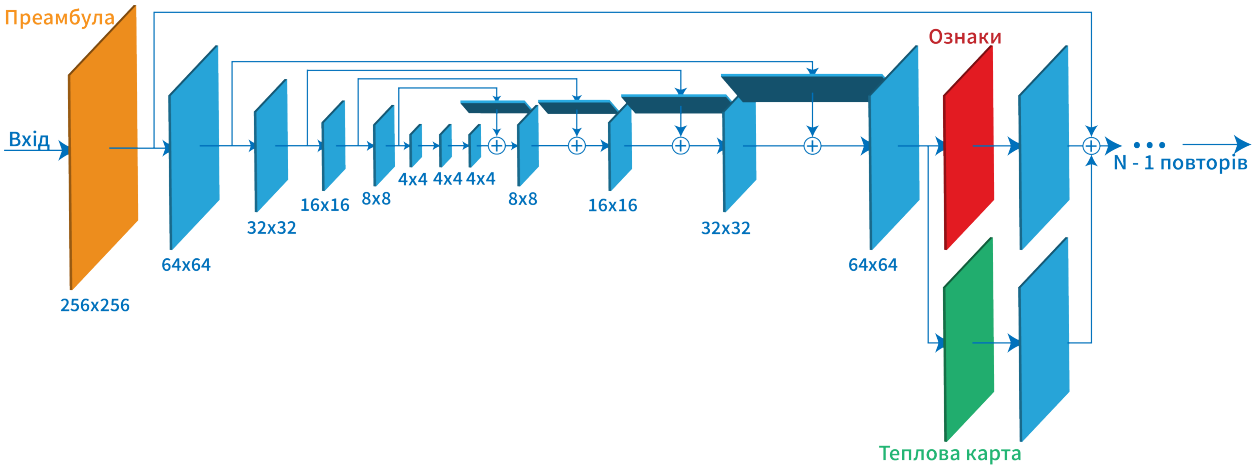


Рис. 1.1. Архітектура Hourglass

цію мережі Hourglass показано на рис. 1.1. Автори відзначають, що обробка зображення в повній роздільній здатності потребувала б занадто багато обчислень і пам'яті, тому на початку мережі згортковий блок (преамбула, показано помаранчевим на рисунку) використовується для швидкої обробки зображення та зменшення роздільної здатності карти ознак до 64×64.

Архітектура допускає *стекування*, тобто модулі Hourglass можуть повторюватися кілька разів послідовно. Зазвичай використовуються стеки з 1, 2 або 4 модулів Hourglass. Після кожного модуля Hourglass формуються карти ознак (показано червоним на рис. 1.1), що подаються в наступний модуль стеку.

Модуль Hourglass відповідає архітектурі кодувальника-декодувальника. Вхідне зображення оброблюється через згорткові блоки з різною роздільною здатністю карт ознак (сині прямокутники на рис. 1.1). Спочатку роздільна здатність карти ознак зменшується після кожного згорткового блоку (частина кодувальника), потім роздільна здатність карт ознак відновлюється (збільшується) після кожного блоку (частина декодувальника). Мережа видає теплові карти з роздільною здатністю 64×64, окрема теплова карта створюється для кожної з ключових точок. Точність оцінки пози людини, визначення ключових точок обличчя, сегментації покращується завдяки обробці зображення в

різних роздільних здатностях.

Загалом багатошарова архітектура Hourglass стає досить глибокою, що сповільнює навчання. Автори пропонують дві ідеї вирішення проблеми: пропускові з'єднання всередині модуля Hourglass та проміжний контроль навчання. По-перше, як чітко видно з рис. 1.1, після кожного згорткового блоку вихід розбивається на дві частини: карти ознак однієї частини зменшуються та передаються в наступний згортковий блок, інша частина «пропускається» від кодувальника до декодувальника. Останню концепцію називають «пропускovим з'єднанням», що вирішує проблему зникаючого градієнту. По-друге, до кожного модуля Hourglass застосовується проміжний контроль навчання (як було сказано раніше, можна створити стеки з кількох модулів Hourglass). Теплові карти прогнозів завжди створюються після кожного модуля Hourglass (показано зеленим кольором на рис. 1.1). Функція втрат при навчанні включає зважену суму втрат для кожного прогнозу теплової карти.

CU-Net [60] намагається покращити архітектуру Hourglass не лише за якістю, але й за використанням пам'яті та часом виконання. Автори відзначають важливість ефективної архітектури для використання на мобільних пристроях. Подібно до Hourglass, мережа приймає зображення 256×256 на вхід та змінює його розмір у преамбулі до 64×64, що залишається максимальним розміром карт ознак до кінця мережі. Аби покращити навчання та забезпечити більш глибокі стеки CU-Net, автори пропонують додати пропускові зв'язки не лише між функціями одного модуля, але й між різними модулями. Щоб уникнути надмірної кількості пропускових з'єднань, введено концепцію зв'язку K-го порядку. Зв'язок K-го порядку означає, що пропускові з'єднання додаються тільки на K модулів вперед. У більшості випадків достатнім є додавання пропускових з'єднань на один модуль вперед ($K = 1$). Автори зменшують споживання пам'яті та покращують швидкість виконання, уникаючи непотрібних копій ознак, спільного використання пам'яті та квантування

Таблиця 1.1. Порівняння кількості параметрів та часу виконання для архітектур Hourglass і CU-Net [60]

Метод	К-сть параметрів (млн)	Виконання (мс)
4×Hourglass	25,5	48,9
8×CU-Net	7,9	36,1
16×CU-Net	15,9	70,8

як ознак, так і параметрів. Крім того, для зменшення загальної кількості параметрів використовуються блоки з меншою кількістю ознак. Усі ці вдосконалення дозволили досягти подібної до Hourglass точності лише з незначною кількістю параметрів і вищою швидкістю виконання. Кількість параметрів і час виконання показані в таблиці 1.1. Можна побачити, що в архітектурі CU-Net більші стеки все ще мають прийнятний час виконання та малу кількість параметрів. Однак, незважаючи на вдосконалення, нещодавно представлені розробки віддають перевагу архітектурі Hourglass над CU-Net.

HRNet [58] також спочатку було запропоновано для оцінки пози людини, а потім адаптовано для задач сегментації, виявлення ключових точок обличчя в [61]. Ця архітектура істотно відрізняється від двох попередніх: HRNet не дотримується архітектури кодувальника-декодувальника та не використовує кілька стеків. Натомість у всій мережі підтримуються паралельні гілки з різною роздільною здатністю ознак. Архітектуру HRNet показано на рис. 1.2. Подібно до попередніх робіт, мережа отримує зображення розміром 256×256, яке потім зменшується до карти ознак розміру 64×64 у преамбулі (на рис. показано помаранчевим). Далі зображення оброблюється за допомогою згорткових блоків та додається ще одна гілка роздільної здатності 32×32. На відміну від попередніх робіт, гілка 64×64 продовжує оброблюватися паралельно. Автори пропонують обмінюватися ознаками між паралельними гілками. Однак, ці карти ознак мають різну роздільну здатність, тому щоб зменшити

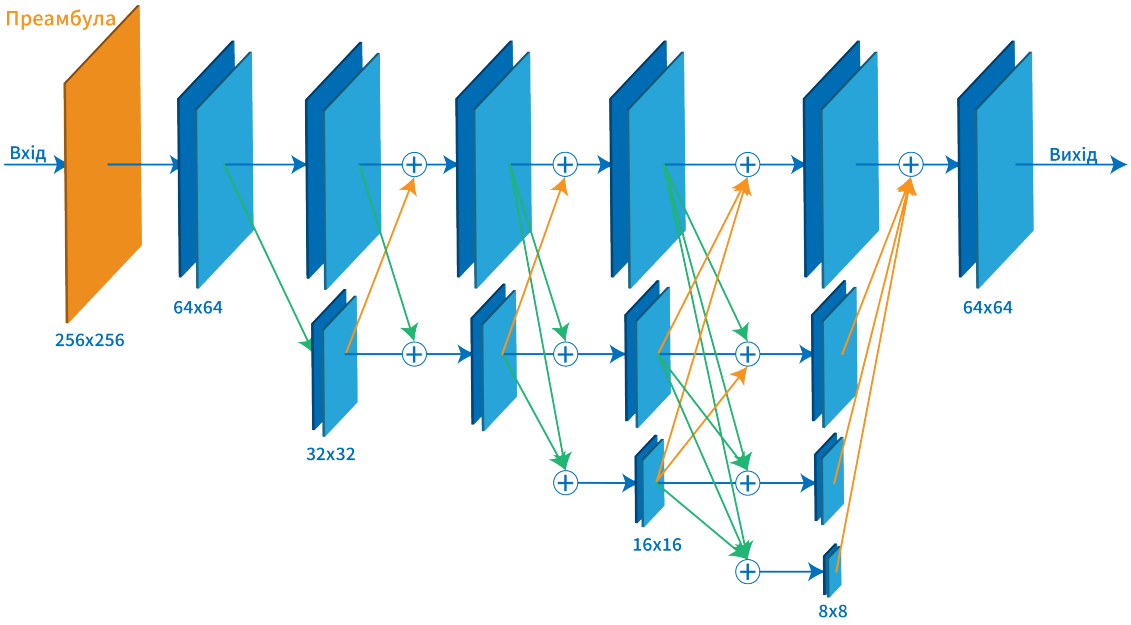
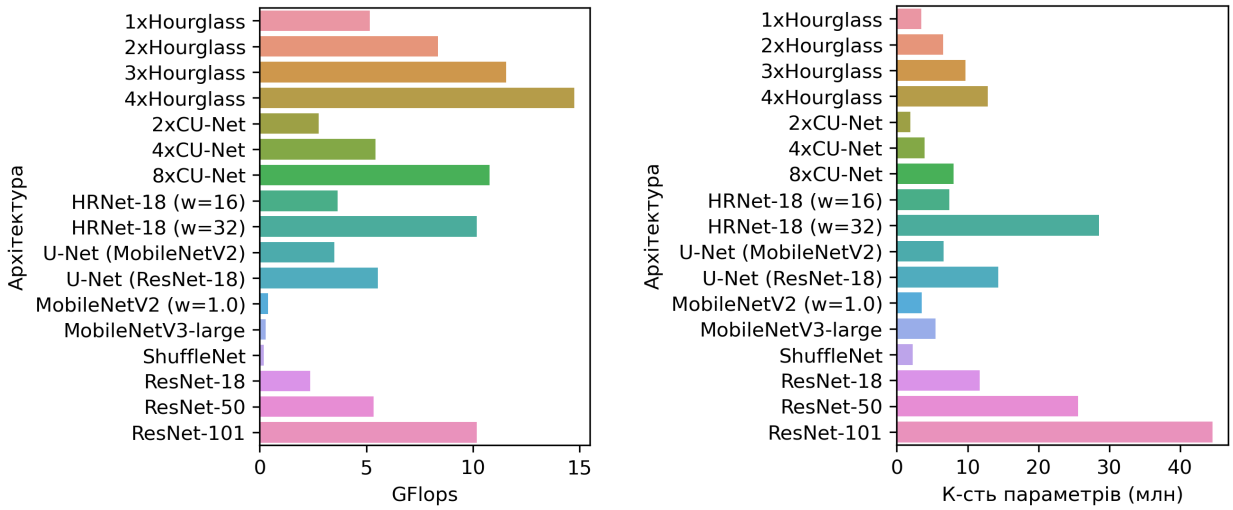


Рис. 1.2. Архітектура HRNet

розмір карти ознак (зелені стрілки) використовується згортка із збільшеним кроком, а щоб збільшити розмір (сині стрілки) – метод найближчого сусіда. До кінця мережі створюються 4 паралельні гілки із різною роздільною здатністю карти ознак. Остаточна теплова карта має суму ознак усіх масштабів, її розмір 64×64. За такої ж кількості параметрів, як і стек із 8 модулів Hourglass, HRNet використовує майже вдвічі менше операцій із плаваючою комою. Крім того, ширину мережі HRNet (кількість згорткових каналів) можна налаштувати для зміни загальної кількості параметрів та швидкості виконання, але лише до початку процедури навчання.

Характеристики розглянутих архітектур підсумовано на рис. 1.3: кількість операцій з плаваючою комою, необхідних для виконання мережі, показано на рис. 1.3а (чим більше число, тим більше часу потрібно для виконання мережі), кількість параметрів на рис. 1.3б (більше параметрів займає більше пам'яті). Використовується позначення $S\times$ для стеку з S модулів, $w = X$ для позначення множника ширини X. Для U-Net в дужках вказано мережу кодувальник. U-Net, Hourglass, CU-Net, HRNet вимагають більше обчислень,



(а) Кількість операцій із плаваючою комою (б) Кількість параметрів

Рис. 1.3. Порівняння архітектур нейронних мереж

ніж інші архітектури, але мають відносно невелику кількість параметрів. Це пояснюється тим, що зазначені мережі враховують вхідні дані із різною роздільною здатністю та створюють карти ознак розміру 64×64 на кожен клас на виході. Інші архітектури (MobileNet, ShuffleNet і ResNet) оброблюють вхідні дані в одному масштабі та мають на виході вектор вірогідностей належності вхідного зображення до певного класу.

Оскільки у даній роботі виконання нейронних мереж здійснюється не тільки на серверних і настільних комп'ютерах, але й на мобільних пристроях, для подальших досліджень обрано архітектуру MobileNetV2 для задачі класифікації, виходячи з її низької обчислювальної складності; мережу U-Net, яка є основою багатьох нейронних мереж, розроблених для задач сегментації та пошуку ключових точок.

1.3 Аналіз практичних застосунків згорткових нейронних мереж

1.3.1 Системи контролю доступу

Багато сучасних підприємств використовують турнікети або розумні двері зі сканерами карт доступу [62–66]. Школи та університети також можуть їх застосувати для забезпечення додаткової безпеки, оскільки вони дешеві та прості у використанні. Переважно використовуються RFID-карти. Водночас такі системи мають серйозний недолік, а саме: картку можна легко втратити, а значить для зловмисника існує можливість непомітно проникнути на підприємство. Це, у свою чергу, може призвести до критичних наслідків, таких як аварія, втрата конфіденційної інформації тощо. Встановлення камер відеоспостереження може бути частковим рішенням, яке дозволить виявити зловмисника заднім числом. Однак тривале зберігання даних відеоспостереження може зайняти багато місця на диску. Найефективнішим, але дорогим, підходом до вирішення проблеми є встановлення біометричних систем розпізнавання обличчя чи відбитків пальців (через термінал або безпосередньо на спеціальній картці контролю доступу).

В статтях [65; 66] розглянуті основні типи мобільних систем контролю доступу, а також переваги таких систем над звичайними RFID картками. Зокрема відмічається зручність використання власного пристрою, відсутність необхідності носити з собою додаткову пластикову карту, можливість повідомити користувача про небезпечну ситуацію шляхом розсилки повідомлень. Розглянуті системи включають продукти, що використовують NFC або Bluetooth Low Energy для передачі інформації на пристрій біля розумної двері. Використання чипа NFC смартфона для безпечної автентифікації викликає дедалі більший інтерес.

Мітка RFID (Radio Frequency Identification) – це пристрій, який може

зберігати невелику кількість даних, зазвичай менше 888 байт (хоча існують модифікації з більшим об'ємом пам'яті, вони рідко використовуються). Смартфони містять чип NFC (Near Field Communication) для читання та програмування міток RFID. Слід зазначити, що не всі існуючі мітки на ринку сумісні з NFC. Три основні типи міток, що підтримують NFC, включають: MIFARE Classic ®, MIFARE Ultralight ® і NTAG ®. Останні два мають найкращу підтримку серед мобільних пристроїв [67], а NTAG має найбільшу ємність. Для розробки системи контролю доступу в розділі 3 буде використано мітки NTAG, а саме дві основі варіанти: NTAG213 (144 байт) і NTAG216 (888 байт) [68]. Окрім пам'яті для запису, мітка також містить серійний номер, можливість увімкнути захист паролем від запису та незворотний перехід у режим лише читання.

Унікальний ідентифікатор у NFC-сумісних системах надається або через глобальний сервер для всіх клієнтів (у цьому випадку стягується абонплата), або безкоштовно на основі унікального ідентифікатора смартфона (IMEI) або SIM-карти (IMSI). Доступ до системи за ідентифікатором можна заблокувати в разі необхідності. При цьому може виникнути щонайменше два випадки несанкціонованого доступу до підприємства, які неможливо відслідкувати: 1) після втрати мобільного пристрою та до блокування його ідентифікатора; 2) у разі умисної передачі смартфона третім особам. Отже, такі системи досить вразливі самі по собі.

Також існують більш просунуті системи, наприклад, у [62] автори відзначають зростаючий інтерес до мультитехнологічних карток із вбудованим сканером відбитків пальців, що є доповненням до стандартної пасивної RFID-мітки. Існують й інші комбіновані системи: патент [63] містить опис біометричної системи, у якій власники RFID-міток також перевіряються за допомогою автономної системи розпізнавання обличчя. Це дозволяє вирішити додаткові проблеми систем контролю доступу, такі як: «відмітка за приятеля»,

коли одна особа реєструє дві картки, а входить лише одна людина; або «притримай двері» – кілька людей заходять, використовуючи одну картку. У випадку порушення, двері будуть замкнені, а спеціальна лампочка попередить охоронців про вторгнення. Подібна система з іншим методом оповіщення запропонована у [64] для контролю доступу в університетських гуртожитках. Звичайно, такі системи мають вищу ціну.

Підводячи підсумок, можна сказати, що всі перераховані вище системи або майже не захищені від передачі карток (наприклад, класичні або системи на основі NFC), або мають високу ціну (біометричні та комбіновані системи). Тому важливою є розробка системи контролю доступу, яка не використовує сканер карток і пропонує більш високі гарантії безпеки при вході через турнікет або розумні двері без використання камер відеоспостереження.

1.3.2 Виявлені проблеми застосунків згорткових нейронних мереж

За результатами аналізу джерел [22; 38; 54–56; 69–89] виявлено наступні проблеми та перспективи подальших досліджень мобільних систем на основі згорткових нейронних мереж:

1. Для пошуку облич [38; 69; 83; 84] співвідношення швидкості виконання та точності потребує покращення – швидші підходи часто мають суттєво нижчу якість. Для навчання моделі потрібен великий анотований набір даних, якщо набір даних є упередженим (незбалансованим) за расою чи статтю, точність виявлення облич у недостатньо представлених групах зазвичай погіршується.

2. Для розпізнавання облич та класифікації емоцій [70; 85–89] проблемами є розпізнавання обличчя, коли фотографії представляють людину різного віку, обличчя сфотографовані під значним кутом та із сильними емоціями та коли обличчя значно перекрите іншими об’єктами. Останнє особливо стало

актуальним, коли медичні маски стали поширеними. Також важливим є покращення базових архітектур нейронних мереж.

3. Для анімації персонажів [71–78]: відео, створене нейронними мережами, не завжди є стабільним у часі, наприклад, анімація обличчя може тремтіти, на екрані можуть з'являтися артефакти. Анімація обличчя, знятого під великим кутом, може призвести до нереалістичної його деформації. Також якість анімації значно погіршується, якщо вихідний персонаж і актор, чиї емоції необхідно відтворити, мають різну форму обличчя.

4. Для ідентифікації стану водія [79–82]: досягнення досить швидкого виконання на мобільному пристрої є проблемою. Відстеження стану водія часто виконується вночі, коли водій погано освітлений, у таких умовах точність визначення стану страждає, особливо з огляду на обмежену обчислювальну потужність.

5. Для систем антиспуфінгу [22; 54–56]: додавання допоміжних вхідних даних до нейронної мережі (наприклад, інформація про глибину, світло або інфрачервоне випромінювання) вимагає додаткових обчислювальних ресурсів або обладнання; проблема антиспуфінгу на мобільних пристроях є недостатньо дослідженою.

Прискорення виконання нейронних мереж, їх швидка та якісна (відповідно до метрик кожної з задач) робота на пристроях із обмеженими можливостями, зокрема мобільних, є однією з суттєвіших проблем для подальших досліджень. Детальний опис підходів для кожної з вищезазначених систем наведено в [20].

1.4 Аналіз методів покращення якості розпізнавання на прикладі пошуку ключових точок

Даний підрозділ містить інформацію про прийоми покращення якості нейро-мережевих методів на прикладі задачі пошуку ключових точок обличчя (іноді в літературі їх називають «орієнтирами обличчя» чи «характеристичними точками», але термін «ключові точки» є найчастіше вживаним). Існуючі оглядові роботи за темою досить старі й здебільшого охоплюють або статистичні методи, або ті, що базуються на ансамблях регресійних дерев [90; 91]. Такі методи демонструють низьку якість визначення ключових точок для знімків, зроблених у необмеженому середовищі. Останнім часом було запропоновано численні підходи на основі нейронних мереж, які показали значно кращу якість. Основну увагу далі зосереджено на нещодавно представлених методах, а саме в 2018–2021 роках. Для повноти також включено деякі важливі старіші методи.

1.4.1 Постановка задачі виявлення ключових точок обличчя

Нехай I буде вхідним зображенням, яке представлено у формі 3-вимірного тензора розміром $W \times H \times C$, де W, H, C – ширина, висота та кількість каналів (кольорів) зображення відповідно. Зазвичай використовуються кольорові зображення з 3 каналами, по одному для червоного, зеленого та синього кольорів. Тоді задача виявлення ключових точок обличчя полягає в тому, щоб знайти таку функцію $\Phi : I \rightarrow Y$, яка на основі вхідного зображення I прогнозує матрицю ключових точок $\hat{Y} \in R^{N_L \times 2}$, де N_L – кількість ключових точок обличчя, $\hat{Y}_{i\,1} \in [0; W)$ відповідає X координаті та $\hat{Y}_{i\,2} \in [0; H)$ відповідає Y координаті i-ї ключової точки. Кількість ключових точок обличчя N_L і точне відображення між i-ю ключовою точкою обличчя та її розташуванням на обличчі (так звана схема анотації) визначаються

на рівні набору даних. Крім того, набір даних визначає, які зображення використовуються для пошуку функції Ф (навчальний набір), а які для її оцінки (тестовий набір).

Далі представлено метрики, які зазвичай використовуються для аналізу якості методів. Зауважимо, що кожен набір даних має спеціальний протокол, який визначає поділ на навчальну та тестову вибірки, метрики для порівняння методів тощо. Основні метрики включають [92; 93]:

1. Нормалізована середня помилка (Normalized Mean Error (NME), %):

$$NME = \frac{1}{K}\sum_{k=1}^{K} NME_k,$$
$$NME_k = \frac{1}{N_L}\sum_{i=1}^{N_L} \frac{\|Y_i - \hat{Y}_i\|}{d} \times 100, \quad (1.1)$$

де Y — матриця правильних ключових точок, $\hat{Y}$ – матриця прогнозованих ключових точок, d – нормалізаційний коефіцієнт (різний для кожного набору даних), N_L – кількість анотацій ключових точок обличчя в наборі даних, K – кількість зображень у тестовому наборі. Нижчі показники кращі.

2. Частота відмов (Failure Rate (FR), %):

$$FR = \frac{1}{K}\sum_{k=1}^{K} [NME_k \geq 10\ \%] \times 100, \quad (1.2)$$

позначає кількість зображень із нормалізованою середньою помилкою, що перевищує порогове значення 10 %. Нижчі показники кращі.

3. Кумулятивна функція розподілу помилок (Cumulative Error Distribution – Area Under Curve (CED-AUC)). Для обчислення даної функції на графіку наноситься частка зображень, нормалізована середня помилка яких менше або дорівнює її значенню на вісі X. Потім обчислюється площа під кривою. Зазвичай NME береться в діапазоні $[0; 10\%]$. Обчислене значення CED-AUC завжди масштабується в діапазоні $[0; 1]$. Із збільшенням

правильно спрогнозованої частини тестового набору збільшується значення CED-AUC.

1.4.2 Новітні методи та ідеї

У цьому підрозділі проведено аналіз нещодавно представлених методів покращення якості виявлення ключових точок обличчя. Інформацію про метод включено, якщо: 1) він покращує найвищу оцінку, встановлену в попередньому році; 2) представлені ідеї потім використовуються в кількох наступних статтях; 3) підхід розширює застосовність методів виявлення ключових точок обличчя або представляє особливу нову ідею, яка раніше не обговорювалася. Якщо подано лише незначну модифікацію, яка не покращує швидкість виконання, якість або застосовність методу, його не включено до цього розділу. Методи було зібрано з різних джерел, включаючи, але не обмежуючись, найпопулярнішими світовими конференціями з комп'ютерного зору. Загалом у цьому розділі проаналізовано 21 метод, для якого наведено як метрики якості, так і часу виконання на центральному, графічному процесорах та смартфоні. За результатами аналізу джерел [94–106] виявлено найпоширеніший та найскладніший набір даних WFLW [94], для якого і наведено метрики якості.

Сучасні методи виявлення ключових точок в необмеженому середовищі засновані на нейронних мережах (рис. 1.4). Вони поділяються на 2 основні категорії: методи *прямої* (або *координатної*) регресії, коли координати *x, y* виявляються моделлю безпосередньо для кожної ключової точки (показано зверху на рисунку); методи регресії на основі *теплової карти*, де двовимірна теплова карта створюється для кожної ключової точки (показано знизу на рисунку). Значення на тепловій карті можна інтерпретувати як імовірність розташування ключової точки в певному місці зображення. Як правило,

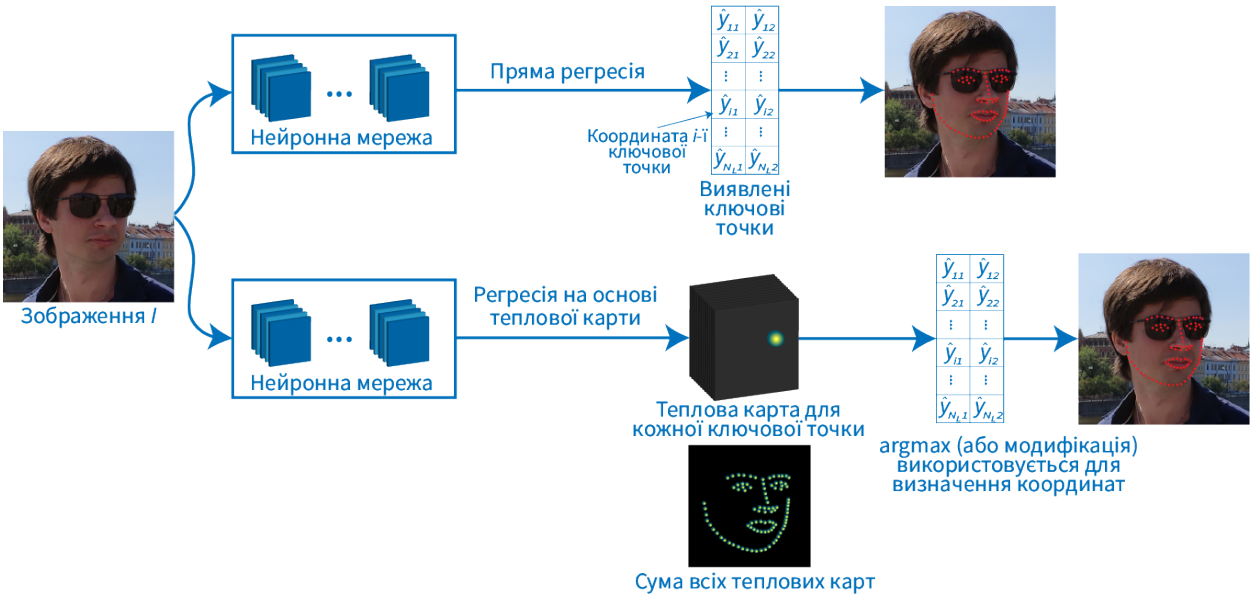


Рис. 1.4. Типи методів виявлення ключових точок

argmax або його модифікація використовується для отримання точних координат ключових точок із теплової карти. Рисунок 1.4 ілюструє обидва підходи. Оскільки архітектури нейронних мереж стають все складнішими, методи зазвичай використовують попередньо визначену архітектуру мережі. Зауважимо, що в багатьох випадках архітектури для визначення ключових точок обличчя та пози людини (всього тіла) однакові. Методи прямої регресії зазвичай використовують широко відомі архітектури зі змагання ImageNet [31], такі як ResNet [40], MobileNetV2 [34], MobileNetV3 [35], ShuffleNet-V2 [107]. Методи на основі теплової карти найчастіше використовують архітектури Hourglass [59], HRNet [58] і CU-Net [60].

В таблиці 1.2 представлено стислу інформацію про типи методів виявлення ключових точок та способи покращення роботи базових архітектур нейронних мереж. Використано такі позначення: D – пряма регресія, H – регресія на основі теплової карти, SM – модель форми, H/D – комбіновані методи, які використовують як теплову карту, так і пряму регресію на різних етапах. Детальний опис кожного з методів наведено в [20].

Таблиця 1.2. Короткий опис методів визначення ключових точок обличчя

Модель	Тип	Основний вклад для задачі виявлення ключових точок
DeFA [108]	SM	3D модель обличчя, знятого під кутом або частково закритого
SAN [109]	H	Модуль нейтралізації стилю
LAB [94]	H/D	Проміжне представлення границь обличчя, яке можна навчати, використовуючи різні схеми анотації
AVS [110]	H	Розширення тренувального набору стилізованими зображеннями
Wing [111]	D	Ф-я втрат, що уточнює прогнози та є нечутливою до викидів
PFLD [112]	D	Нова функція втрат. Запропоновано мобільну реалізацію
FAN [113]	H	Бінаризовані згортки
AWing [93]	H	Уточнення теплових карт за допомогою спеціальної ф-ї втрат
MobileFAN [114]	H	Дистиляція нейронних мереж
GEAN [115]	H	Розширення тренувального та тестового наборів змагальними атаками
HRNetV2 [61]	H	Покращення архітектури HRNet
LUVLi [92]	H	Оцінка впевненості виявлення для кожної з ключових точок
DAG [116]	D	Каскад графових нейронних мереж. Вивчає структурну інформацію обличчя
PropagationNet [117]	H	Модуль уваги меж обличчя, ф-я втрат Focal Wing. Кращий результат на 300W
SAAT [118]	H	Використання GAN і змагальних атак для генерації тренувальних зображень
LDDMM-Face [119]	SM	Тренування на розрідженій анотації, виявлення щільної розмітки ключових точок
AnchorFace [120]	D	Виявлення опорної точки та наступне її уточнення
PIPNet [121]	H/D	Уточнення грубої теплової карти модулем регресії
ADNet [122]	H	Теплові карти Точка-Границя. Кращі результати на декількох наборах
HIH [123]	H	Зменшення помилки квантування за допомогою вкладених теплових карт
Subpixel Heatmap [124]	H	Зменшення помилки квантування за допомогою локального м’якого argmax. Кращі результати на декількох наборах

В таблиці 1.3 зазначено архітектури нейронних мереж, які використовуються в методах виявлення ключових точок, наведено оцінки кількості параметрів, операцій множення-додавання та часу виконання на центральному, графічному процесорі комп'ютера та на смартфоні. Представлені дані в таблицях зібрано з відповідних статей. Для методів SAN, LAB, Wing, AVS, AWing додатково використано джерела даних [112; 114; 117; 120; 121; 123]. Кращий результат за кількістю параметрів і операцій множення-додавання показано червоним, другий – синім. Зеленим кольором виділено час виконання менше 17 мс (60 кадрів на секунду). Окрім розпізнавання ключових точок обличчя зазвичай виконуються й інші методи, тому поріг такий суворий. Архітектури Hourglass і CU-Net зазвичай складаються в стек. $N\times$Hourglass використовується для позначення стека N модулів Hourglass. Кількість параметрів перетворюється на споживання пам'яті пристрою, що особливо важливо для мобільних та крайових пристроїв. Кількість операцій множення додавання, необхідних для виконання мережі, визначає вимоги до швидкодії процесора пристрою. Оцінка часу виконання методів наведена відповідно до результатів, зазначених авторами. Заміри часу виконувались на дещо різному обладнанні, тим не менш, наведені результати дозволяють оцінити порядок швидкодії. Відмітимо, що більшість авторів надають інформацію лише про швидкість виконання на графічному пристрої, а для 11 із 21 розглянутих методів взагалі не надають дані про швидкодію. Незважаючи на те, що підходи з найвищими якісними метриками мають велику кількість операцій множення-додавання, все ще існують легкі моделі, які є досить точними. Основа Hourglass є найпоширенішою серед сучасних методів виявлення ключових точок (використовується в 9 із 21 випадку). Лише один метод (PFLD) був адаптований до мобільного пристрою і може працювати на ньому в реальному часі. Загалом, пришвидшення методів виявлення ключових точок, зокрема для мобільних пристроїв є важливим напрямком подальших дослі-

Таблиця 1.3. Обчислювальна складність та швидкість виконання методів виявлення ключових точок та архітектур нейронних мереж в їх основі

Метод	Архітектура	Парам. (млн)	Оп. мн.-дод. (млрд)	Час виконання (мс)		
				ЦП	ГП	Моб.
DeFA	CNN	-	-	-	-	-
SAN	ResNet-152	-	-	-	343	-
LAB	4×Hourglass	25.1	18.85	2600	60	-
Wing	CNN-6	3.8	-	6.7	2.5	-
Wing	CNN-6/7	12.3	-	50	5.9	-
Wing	ResNet-50	25	5.5	125	33.3	-
AVS	ResNet-152	35.02	33.87	-	-	-
PFLD 0.25X	MobileNetV2	-	-	1.2	1.2	7.0
PFLD 1X/1X+	MobileNetV2	-	-	6.1	3.5	26.4
FAN	4×Hourglass	24	-	-	33.3	-
AWing-1HG	1×Hourglass	-	-	-	8.3	-
AWing-2HG	2×Hourglass	-	-	-	15.7	-
AWing	4×Hourglass	24.15	26.79	-	29.0	-
MobileFAN (0.5)	MobileNetV2	1.84	0.45	-	4.0	-
MobileFAN	MobileNetV2	2.02	0.72	-	4.2	-
GEAN	4×Hourglass	-	-	-	58.8	-
HRNetV2	HRNet-18	9.3	4.3	-	-	-
LUVLi	8×CU-Net	-	-	-	17	-
DAG	HRNet-18	-	-	-	-	-
PropagationNet	4×Hourglass	36.30	42.83	-	-	-
SAAT	2×Hourglass	-	-	-	-	-
LDDMM-Face	HRNet-18	-	-	-	-	-
AnchorFace	ShuffleNet-V2	-	1.71	-	22.2	-
AnchorFace	HRNet-18	-	5.30	-	-	-
PIPNet	MobileNetV2	4.2	0.5	33.9	8.3	-
PIPNet	MobileNetV3	4.5	0.4	35.2	12.5	-
PIPNet	ResNet-18	12.0	2.4	28.0	5.0	-
PIPNet	ResNet-101	45.7	10.5	113.6	17.9	-
ADNet	4×Hourglass	13.37	17.04	-	95.29	-
HIH_C	2×Hourglass	14.47	10.38	-	-	-
HIH_T	2×Hourglass	28.18	10.29	-	-	-
SubpixelHeatmap	2×Hourglass	-	-	-	-	-

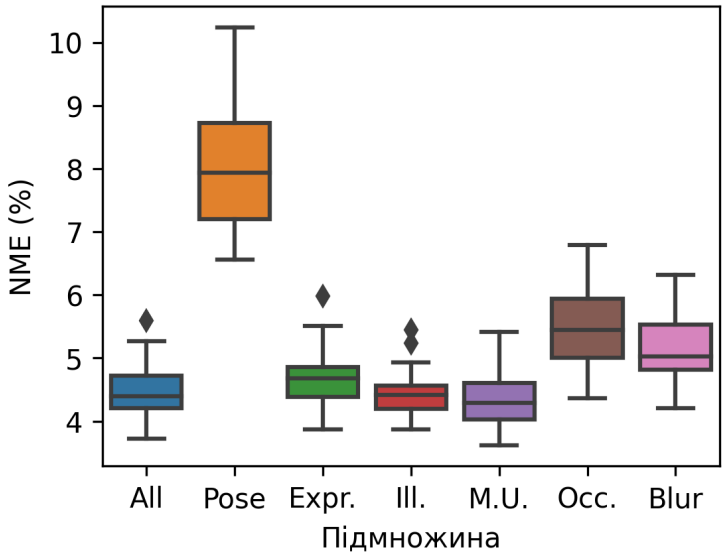


Рис. 1.5. Порівняння NME на підмножинах набору даних WFLW

джень.

Для аналізу та порівняння архітектур нейронних мереж, впливу складних категорій зображень на результуючу якість розпізнавання розглянуто набір даних виявлення ключових точок WFLW [94]. Якісні метрики для методів виявлення ключових точок представлено в таблиці 1.4. Показано NME (%), частоту відмов (FR, %) і CED-AUC (позначено як AUC у таблиці) для всього тестового набору. Показано помилки для всіх типів складних категорій зображень, присутніх у наборі даних WFLW: незвична поза (Pose), вираз (Expr.), освітленість (Ill.), макіяж (M.U.), оклюзія (закриття обличчя, Occ.) і розмиття (Blur). Найкращий результат показано червоним кольором, другий – синім.

На рис. 1.5 показано коробкові діаграми нормалізованої середньої помилки (NME, %) на наборі даних WFLW для методів із табл. 1.4. Результати показано для всього тестового набору (All), а також для підмножин, які зосереджуються на незвичайній позі (Pose), виразі (Expr.), освітленості (Ill.), макіяжі (M.U.), оклюзії (Occ.) і розмитості (Blur). Відзначено значну різницю в NME для різних підмножин. Найбільш суттєвою проблемою для наборів даних виявлення ключових точок є поза (найкраща помилка все ще становить

Таблиця 1.4. Якість виявлення ключових точок на наборі даних WFLW

Метод	Тестовий набір			Піднабори (NME %, ↓)					
	NME %, ↓	FR %, ↓	AUC ↑	Pose	Expr.	Ill.	M.U.	Occ.	Blur
LAB	5.27	7.56	0.5323	10.24	5.51	5.23	5.15	6.79	6.32
Wing (тест з [93])	5.11	6.00	0.5504	8.75	5.36	4.93	5.41	6.37	5.81
AVS	4.39	4.08	0.5913	8.42	4.68	4.24	4.37	5.60	4.86
AWing	4.36	2.84	0.5719	7.38	4.58	4.32	4.27	5.19	4.96
MobileFAN (0.5)	5.59	6.72	0.4682	9.68	5.98	5.45	5.33	6.49	6.31
MobileFAN	4.93	5.32	0.5296	8.72	5.27	4.93	4.70	5.94	5.73
HRNetV2	4.60	-	-	7.94	4.85	4.55	4.29	5.44	5.42
LUVLi	4.37	3.12	0.577	-	-	-	-	-	-
DAG	4.21	3.04	0.5893	7.36	4.49	4.12	4.05	4.98	4.82
PropagationNet	4.05	2.96	0.6158	6.92	3.87	4.07	3.76	4.58	4.36
LDDMM-Face	4.63	3.68	0.5509	-	-	-	-	-	-
AnchorFace	4.62	4.20	0.5516	-	-	-	-	-	-
$\text{AnchorFace}_{(\text{HRNet-18})}$	4.32	2.96	0.5769	-	-	-	-	-	-
SAAT	5.11	5.63	0.5633	-	-	-	-	-	-
$\text{PIPNet}_{(\text{MobileNetV2})}$	4.79	-	-	8.76	4.86	4.56	4.60	6.04	5.53
$\text{PIPNet}_{(\text{MobileNetV3})}$	4.65	-	-	8.22	4.75	4.49	4.46	5.72	5.31
$\text{PIPNet}_{(\text{ResNet-18})}$	4.57	-	-	8.02	4.73	4.39	4.38	5.66	5.25
$\text{PIPNet}_{(\text{ResNet-101})}$	4.31	-	-	7.51	4.44	4.19	4.02	5.36	5.02
ADNet	4.14	2.72	0.6022	6.96	4.38	4.09	4.05	5.06	4.79
ADNet (focal loss)	3.98	2.00	0.6250	6.56	4.02	3.87	3.62	4.36	4.21
HIH_C	4.18	2.96	0.597	7.20	4.19	4.45	3.97	5.00	4.81
HIH_T	4.21	2.84	0.593	7.20	4.28	4.42	4.03	5.00	4.79
SubpixelHeatmap	3.72	1.55	0.631	-	-	-	-	-	-

6,56 %), потім йдуть оклюзія (4,36 %) і розмиття (4,21 %). В той самий час, макіяж (3,62 %), освітленість (3,87 %) і вираз (3,87 %) складають найменшу проблему. Незважаючи на те, що протягом останніх 3 років було досягнуто покращення в 1,6 для підмножини з незвичною позою, необхідне подальше вдосконалення методів виявлення ключових точок.

За результатами аналізу джерел [50; 61; 77; 90–94; 108–134] виявлені такі способи підвищення точності нейро-мережевих методів для задачі виявлення ключових точок:

— використання допоміжного представлення, яке містить структурну інформацію про обличчя, наприклад: об'ємної сітки обличчя (DeFA); моделі деформованої форми (LDDMM-Face); проміжного представлення на основі графів (LAB); кутів повороту об'ємної моделі обличчя (PFLD); видимості ключових точок (LUVLi); представлення обличчя як графової моделі (DAG); регресії зміщення відносно опорних точок, визначених для облич із різними позами (AnchorFace) або регресії зміщення від сусідніх ключових точок (PIP);

— представлення границь обличчя явно (LAB) або через модуль уваги (PropagationNet, ADNet);

— вибір складних прикладів під час навчання. Різні варіації на тему були представлені в роботах Wing, PFLD і PropagationNet;

— агрегування прогнозів для кількох вхідних зображень: із модифікацією стилю (SAN); після змагальної атаки (GEAN); або з кількома різними розширеннями вхідного зображення (SubpixelHeatmap);

— розширення тренувального набору за допомогою стилізації зображень (AVS) або змагальних атак (SAAT);

— зменшення вкладу викидів в функції втрат та збільшення внеску малих і середніх помилок (для кращого уточнення прогнозів): Wing, AWing і похідні роботи;

— субпіксельна точність на тепловій карті (зменшення помилки квантування теплової карти): зважений argmax (LUVLI), спільна теплова карта та пряма регресія (PIPNet), глобальний м'який argmax (ADNet), уточнення на основі згорткових мереж або трансформерів (HIH), локальний м'який argmax (SubpixelHeatmap).

1.5 Методи навчання нейронних мереж за кількома прикладами

Іншою важливою задачею при навчанні нейронних є необхідність зменшення кількості прикладів, необхідних для навчання мережі та її узагальнення на нові, не бачені при тренуванні дані. Наприклад, всі конфігурації нейронних мереж ResNet, MobileNetV2, MobileNetV2, ShuffleNet тощо навчаються на наборі даних ImageNet [31], який містить 1 281 167 зображень і 1000 класів (близько 1200 зразків на клас). Очевидно, що для багатьох практично важливих задач неможливо зібрати та розмітити такий великий набір даних, а навчання випадково-ініціалізованої мережі на малому наборі даних призведе до поганих результатів.

Щоб підвищити якість навчання на менших наборах даних, можна використати підхід так званого «трансферного навчання». Як правило, береться мережа, попередньо навчена на наборі даних ImageNet і використовуючи отримані ваги, мережу навчають ще кілька епох на цільовому (меншому) наборі даних [40; 135; 136]. Однак, навчання мережі, коли на кожен клас є лише поодинокі приклади (наприклад, 1, 2, 5, 10 прикладів на клас), все ще є проблемою. Це відрізняється від того, як навчаються люди, коли навіть одного прикладу, поданого дитині, може бути достатньо, щоб навчитись відрізняти деякий об'єкт. Крім того, із використанням трансферного навчання важко апріорно оцінити якість попередньо навченої на ImageNet мережі на цільовому наборі даних. Отже, отримуємо проблему вибору моделі: якщо модель A

краща за модель B на ImageNet, чи буде вона кращою на невеликому наборі даних?

Багатообіцяючим підходом до вирішення проблем зменшення кількості навчальних прикладів та апріорної оцінки можливості узагальнення мережі до нових задач є використання методів мета-навчання, зокрема підходу навчання за кількома прикладами (few-shot learning) [137–140]. Методи мета-навчання тренують мережу на різних «задачах», які випадковим чином вибираються з усього простору задач. Навчаючи мережу таким чином, очікується, що мережа вивчатиме ознаки, спільні для всіх задач, а не лише одного.

Застосунки мета-навчання існують буквально в усіх сферах машинного навчання [137–140], таких як обробка природної мови, навчання з підкріпленням, верифікація облич тощо. Підходи до мета-навчання в основному поділяються на 3 широкі категорії [141]: метричні методи, модельні методи, оптимізаційні методи. Представники кожної групи відрізняються підходом до побудови архітектури нейронної мережі та процедурою навчання:

1. Метричні методи будують архітектуру нейронної мережі, виходом якої є вектор ознак в метричному просторі, та визначають міру подібності (метрику) в даному просторі. Відстань між векторами екземплярів одного класу має бути меншою, ніж між представниками різних класів. До таких методів відносять сіамські мережі [142], мережі відповідності [143], мережі прототипів [144].

2. У модельних методах архітектуру мережі розроблено так, щоб вона мала явні комірки пам’яті, які допомагають мережі швидко адаптуватися. Метод нейронних мереж з доповненою пам’яттю [145] є яскравим прикладом цього підходу.

3. Оптимізаційне навчання реалізується через зміну процедури навчання і функції втрат, та не передбачає зміни архітектури.

Одним з найвідоміших та ключових методів оптимізаційного мета-

навчання є мета-навчання довільної моделі (**MAML**) [137]. В зазначеній роботі показано застосування методу в регресії, класифікації та навчанні з підкріпленням. Для класифікації зображень було розглянуто два популярні набори даних: Omniglot [146] і miniImageNet [143; 147], де MAML з перевагою переміг багато попередніх методів. Для експериментів на miniImageNet автори MAML визначили архітектуру згорткової нейронної мережі «CNN4». Велика кількість подальших методів використовують її та метод MAML в якості базових, пропонуючи модифікації методу навчання. **Reptile** [148] спростив схему навчання MAML, **MAML++** [149] містить практичні рекомендації щодо покращення стабільності навчання. Авторами було відмічено, що хоча MAML++ додав більше параметрів до мережі, загальний час навчання зменшився завдяки запропонованим модифікаціям методу. На думку авторів **Meta-SGD** [150] шляхом вивчення не лише параметрів мережі, але й окремих коефіцієнтів оновлення для кожного з параметрів, можна досягти вищої точності класифікації. Однак при такому підході час навчання мережі та споживання пам’яті значно зростають, оскільки виникає необхідність оптимізувати вдвічі більше параметрів. За результатами аналізу [137; 148; 150; 151] виявлено, що процедура адаптації до нових класів оптимізаційних методів навчання за кількома прикладами виконується досить повільно, а її прискорення є актуальною науковою задачею.

Висновки до розділу 1

В даному розділі розглянуто проблеми впровадження нейронних мереж в наступних застосунках: пошук та ідентифікація облич, розпізнавання емоцій, антиспуфінг, ідентифікація стану водія, анімація персонажів, пошук ключових точок. Проведено порівняльний аналіз архітектур нейронних мереж:

— для задач класифікації: AlexNet, VGG, ResNet, MobileNetV2, SENet, MnasNet, MobileNetV3;
— для задач сегментації: U-Net, Hourglass, HRNet, CU-Net.

Для кожної мережі наведено її особливості, оцінено обчислювальну складність (за кількістю операцій множення та додавання) та розраховано кількість параметрів; проаналізовано переваги та недоліки. Розглянуто існуючі методи прискорення навчання та виконання нейронних мереж.

За проведеним аналізом застосунків мобільних нейронних мереж зазначено важливість зменшення часу їх виконання, не втрачаючи якості розпізнавання зображень, та зміни конфігурацій нейронних мереж під час їх розгортання на пристроях із різними обчислювальними можливостями.

Також проаналізовано методи мета-навчання, що дозволяють навчити мережу лише за кількома прикладами на клас. MAML є ключовим методом оптимізаційного мета-навчання та є основою великої кількості подальших підходів. Виявлено, що недоліком таких методів є повільна процедура адаптації мережі до нових класів.

Для досягнення мети дисертаційної роботи на основі проведеного в даному розділі аналізу було вирішено для подальших досліджень обрати:

— мережу MobileNetV2, котра широко використовується для вирішення багатьох практичних проблем комп’ютерного зору, зокрема задачі класифікації;
— мережу U-Net, яка є основою багатьох нейронних мереж, розроблених для задач сегментації;
— мережу CNN4, яка є основою методів оптимізаційного навчання за кількома прикладами, базовим з яких є метод MAML.

РОЗДІЛ 2
ЗМІНЮВАНА ЗГОРТКОВА НЕЙРОННА МЕРЕЖА ДЛЯ ЗАДАЧІ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ТА МЕТОД ЇЇ НАВЧАННЯ

Даний розділ присвячено розробці архітектури змінюваної згорткової мережі та методу навчання такої мережі для класифікації зображень. Для цього вирішені наступні задачі:

1. Запропоновано архітектуру змінюваної згорткової нейронної мережі для задачі класифікації із вбудованим в неї згортковим блоком, що може змінюватись в процесі або після навчання, та метод навчання такої мережі для ефективного виконання (співвідношення якість/час) на пристроях з різними, значно відмінними між собою обчислювальними потужностями.
2. Здійснено програмну реалізацію запропонованого методу та впроваджено розроблений блок в архітектуру нейронної мережі MobileNetV2.
3. Проведено експериментальні дослідження на загальновизнаному наборі даних класифікації ImageNet, порівняно розроблену архітектуру змінюваної нейронної мережі з іншими провідними архітектурами нейронних мереж. Проведено дослідження на наборі антиспуфінгу CelebA-Spoof. Результати досліджень підтвердили ефективність запропонованого методу.

2.1 Постановка задачі класифікації. Задача антиспуфінгу

Нехай X – вхідне зображення, яке представлено у формі 3-вимірного тензора розміром $W \times H \times C$, де W, H, C – ширина, висота та кількість каналів (кольорів) зображення відповідно. Зазвичай використовуються кольорові зображення з 3 каналами, по одному для червоного, зеленого та синього ко-

льорів. Тоді задача класифікації зображень полягає в тому [152], щоб знайти таку функцію $\Phi : X \to \{1 \dots D\}$, яка за вхідним зображенням X прогнозує номер класу $\hat{d} \in 1 \dots D$, до якого належить вхідне зображення, де D – кількість класів. Часто зустрічається формулювання, коли результатом функції Φ є вектор $\hat{y}$ із розподілом ймовірностей належності зображення до класів. Після цього обчислюється $\hat{d} = \text{argmax}_i \hat{y}_i$. Для навчання функції Φ використовуються анотовані набори даних із відомим істинним номером класу d, а істинний унітарний вектор y формується так: $\forall i \neq d : y_i = 0, y_d = 1$.

В рамках одного набору даних для навчання та тестування нейронної мережі використовуються різні зображення для оцінки узагальнюючої можливості мережі. Для оцінки якості зазвичай використовується точність або інші метрики відповідно до вирішуваної проблеми в рамках задачі класифікації. Найбільш широко використаними наборами даних для класифікації зображень є MNIST [153], CIFAR [154], ImageNet [31]. ImageNet є визнаним стандартним набором даних для навчання та оцінки якості нових архітектур нейронних мереж, що містить 1281167 зображень для навчання мережі та 50000 для тестування. Кожне зображення анотоване одним з 1000 класів.

Однією із задач класифікації є антиспуфінг обличчя, мета якого полягає у перевірці чи є обличчя перед камерою справжнім обличчям людини, яка стоїть, чи фотографією, яку тримають, щоб обдурити систему. Антиспуфінг використовується для вдосконалення біометричних систем контролю доступу для кращого виявлення несанкціонованого доступу. Проблема антиспуфінгу ускладнюється великою кількістю способів спуфінгу, таких як надруковане зображення обличчя, плакат, відео, маска для обличчя тощо. Залежно від архітектури системи контролю доступу, перевірка має здійснюватися на сервері або безпосередньо на мобільному пристрої (наприклад, [22]). Недоліки існуючих методів антиспуфінгу зазначено в розділі 1.3.2, а їх опис в [20].

2.2 Блок PTA змінюваної згорткової мережі

Початкові дослідження згорткових нейронних мереж було зосереджено виключно на якості, незалежно від витрат на обчислення, але сучасним трендом є розробка швидких і в той самий час точних нейронних мереж, що зумовлено вимогами малої затримки обробки, конфіденційності даних користувача, роботи без Інтернету та зменшення навантаження на сервер. Крім того, що мобільні пристрої та пристрої Інтернету речей мають значно меншу обчислювальну потужність, зазвичай слід враховувати кілька поколінь або цінових категорій таких пристроїв. В розділі 1 проаналізовано архітектури нейронних мереж та методи їх вдосконалення, що зустрічаються в застосунках, але всі вони вимагають остаточного визначення архітектури перед навчанням. Можливість змінювати конфігурацію мережі після її навчання відсутня. Існує два варіанти застосування нейронних мереж на пристроях із різними обчислювальними ресурсами:

— навчати окрему мережу для кожної категорії пристроїв, що потребує більше часу та зусиль;

— розробити єдину архітектуру, яку було б націлено на просунуті пристрої та якість, або на бюджетні (із малими обчислювальними можливостями) та швидкість. В першому випадку, на бюджетних пристроях виконання буде надто повільним, а в другому – ресурси просунутих пристроїв не будуть застосовані повною мірою, а якість буде нижчою за можливу на таких пристроях.

Обидва ці рішення не є оптимальними.

Для вирішення проблем, пов’язаних з розгортанням нейронних мереж на пристроях із різними обчислювальними характеристиками, у цій роботі запропоновано архітектуру змінюваної згорткової нейронної мережі для задачі класифікації із вбудованим в неї згортковим блоком PTA, що може змінюва-

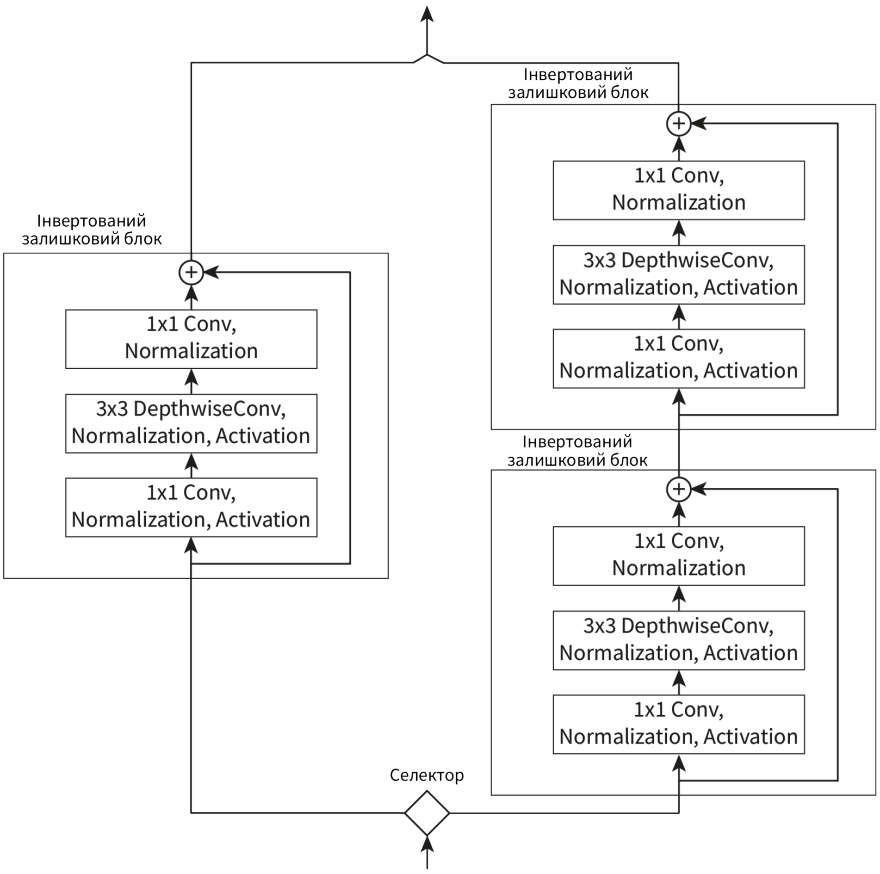


Рис. 2.1. Схема блоку PTA

тись в процесі або після навчання, та метод навчання такої мережі. Далі цей додаткових блок будемо стисло називати блок PTA, а саму згорткову мережу із таким блоком – мережею PTA.

Запропонований блок PTA має просту структуру (зображено на рис. 2.1) та є заміною пари інвертованих залишкових блоків архітектури MobileNetV2. Блок має 2 гілки: важку (на рис. праворуч) та легку (ліворуч). Важка є повністю еквівалентною парі інвертованих залишкових блоків, а легка є вдвічі швидшою та виконує один інвертований залишковий блок. Гілка, яку потрібно виконати, динамічно обирається на основі конфігурації користувача під час виконання. Виконувати можна кожну гілку окремо або обидві одночасно. Якщо виконуються обидві гілки, їхні виходи усереднюються поелементно для того, щоб розподіл ознак залишався незмінним. Кожна гілка має власні, не спільні з іншою, ваги. Один інвертований залишковий блок має 3

Таблиця 2.1. Запропонована архітектура мережі PTA
для задачі класифікації

Тип блока	Каналів	Повторів	Крок
Інвертований залишковий	16	1	1
Інвертований залишковий	24	2	2
Інвертований залишковий	32	3	2
Інвертований залишковий	64	2	2
Блок PTA	64	1	1
Інвертований залишковий	96	1	1
Блок PTA	96	1	1
Інвертований залишковий	160	1	2
Блок PTA	160	1	1
Інвертований залишковий	320	1	1

згорткових шари, а отже, в залежності від конфігурації блоку PTA, він може виконувати 3, 6 або 9 згорткових шарів при використанні легкої, важкої або обох гілок відповідно.

В архітектурі MobileNetV2 було замінено три пари інвертованих залишкових блоків з найбільшою кількістю каналів на блоки PTA, як показано в табл. 2.1, де зазначено тип блоку, кількість вихідних каналів, кількість повторів блоку та крок першої згортки блоку. Завдяки блокам PTA мережа дозволяє змінювати кількість згорткових шарів під час або після навчання від 43 (для конфігурації з усіма легкими гілками) до 61 (для конфігурації, де у всіх блоках використовуються обидві гілки). Для порівняння оригінальна мережа MobileNetV2 має 52 згорткових шари, кількість або характеристики яких не може бути змінено після навчання.

Якщо h, w – висота та ширина входу відповідно, d', d'' – кількість вхідних та вихідних каналів, $t = 6$ – параметр розширення, $k = 3$ – розмір ядра згортки, що розділяється, то асимптотична складність виконання PTA блоку є величиною $O(h \cdot w \cdot d' \cdot t(d' + k^2 + d''))$. Отже, блок PTA має один і той самий

Таблиця 2.2. Апріорні частоти вибору
конфігурацій блоків PTA під час навчання

Конфігурація	Частота вибору
PTA-HHH	0,45
PTA-LHH	0,15
PTA-HLH	0,15
PTA-HHL	0,15
PTA-LLL	0,10

порядок складності, що й інвертований залишковий блок, обчислений [34].

Для задачі класифікації використано перехресну ентропію як функцію втрат. Розглянута мережа виводить логіти, а отже перехресна ентропія також включає обчислення softmax і визначається наступним чином:

$$\mathcal{L}(y, \hat{y}) = -\sum_{n=1}^{N}\sum_{d=1}^{D} y_{n,d} \log \frac{\exp(\hat{y}_{n,d})}{\exp(\sum_{i=1}^{D} \hat{y}_{n,i})}, \tag{2.1}$$

де N – це кількість зразків у міні-пакеті, $D = 1000$ – кількість класів, $\hat{y}_{n,d}$ – вихідний логіт нейронної мережі Ф для елемента n та класу d, $y_{n,d}$ – відповідне правильне (розмічене) значення.

Під час навчання такої мережі конфігурація блоків PTA обирається випадково на кожній ітерації, ваги мережі оновлюються лише для неї. Щоб уникнути надмірної випадковості в мережі, кількість можливих конфігурацій блоків обмежено п’ятьма:

— усі блоки виконують важку гілку;
— один з блоків виконує легку, інші – важку (3 можливі комбінації);
— всі блоки виконують легку гілку.

Вибір конфігурації може виконуватись за таких міркувань: чим більше параметрів в конфігурації, тим частіше така конфігурація має обиратися під час навчання. Апріорний розподіл частот показано в табл. 2.2. Конфігурація вказується для кожного з 3 блоків PTA, для позначення використовується

3-літерна абревіатура. Літери H, L, B використовуються для позначення виконання важкої, легкої та обох гілок відповідно для кожного з блоків PTA. За апріорними частотами обидві гілки не виконуються одночасно, щоб не сповільнювати процес навчання, тим не менш в експериментах далі буде продемонстровано, що така конфігурація може бути ефективно використаною для класифікації зображень. Всі конфігурації, яких немає в табл. 2.2, ніколи не обираються під час навчання.

Алгоритмічна реалізація однієї епохи методу навчання змінюваної згорткової мережі виглядає наступним чином:

1. Для кожного міні-пакету $\{X, y\}$:
2. Випадково обрати конфігурацію PTA відповідно до розподілу з табл. 2.2.
3. Розрахувати функцію втрат за формулою (2.1).
4. Оновити параметри θ мережі Φ, що відповідають обраній конфігурації PTA.

2.3 Експериментальні результати виконання змінюваної мережі на наборі даних класифікації зображень ImageNet

В даному підрозділі проведено експериментальне дослідження часу виконання та якості мережі PTA на наборі даних ImageNet, що є стандартним набором даних для звітування результатів нових архітектур нейронних мереж [32; 34–37; 40; 107; 135; 136; 155]. Базова конфігурація навчання заснована на запропонованій в вищезазначених роботах. Процедура обробки зображення: з зображення вирізається випадковий фрагмент розміру 208×208, далі використано розширення тренувального набору: горизонтальне віддзеркалення, TrivialAugment [156], CutMix [157] та mixup [158]. Отримане зображення нормалізується. Методом навчання є стохастичний градієнтний спуск,

крок навчання обирається відповідно до стратегії CosineAnnealing [159] із максимальним кроком навчання $\alpha_{max} = 0.5$ та мінімальним $\alpha_{min} = 3 \cdot 10^{-5}$. Модель навчається 200 епох із розміром міні-пакету 256. Використовується L_2 регуляризація з множником 10^{-5}. Навчання мережі PTA відбувається із змішаною точністю (fp16), тестування із повною – fp32. Для навчання змінюваної згорткової мережі використовується стратегія вибору конфігурацій PTA, як описано в розділі 2.2.

Результати показано на тестовому наборі, для якого зображення зменшуються до розміру 232×232 із білінійною інтерполяцією, далі проводиться центральна обрізка зображення до 224×224 та нормалізація. Точність інших нейронних мереж для порівняння на наборі даних ImageNet взято з [160].

Процедуру навчання реалізовано мовою програмування Python 3 із використанням бібліотеки PyTorch [161]. Для оцінки часу виконання на пристроях із різними обчислювальними можливостями реалізовано застосунок MLBench мовою програмування Kotlin для операційної системи Android та мовою C++ для крайових пристроїв, виконання на центральному та графічному процесорі комп'ютера. У всіх випадках для розгортання на пристрої нейронної мережі використовується серіалізована модель, що виконується в libTorch [161].

Для оцінки якості виконання мережі на ImageNet використовуються метрики точність: (топ 1) (2.2) та точність (топ 5) (2.3):

$$\text{Accuracy (Top 1)} = \frac{1}{N}\sum_{n=1}^{N} 1(d = \underset{i=\overline{1,D}}{\text{argmax}}\hat{y}_{n,i}) \tag{2.2}$$

$$\text{Accuracy (Top 5)} = \frac{1}{N}\sum_{n=1}^{N} 1(d \in \underset{i=\overline{1,D}}{\arg\,\text{top}_5}\hat{y}_{n,i}) \tag{2.3}$$

де функція $\arg\,\text{top}_5$ повертає індекси 5 найбільших елементів у векторі.

Для відтворюваності результатів всі експерименти проводяться із використанням псевдо-випадкового генератора чисел із фіксованим зерном, що є

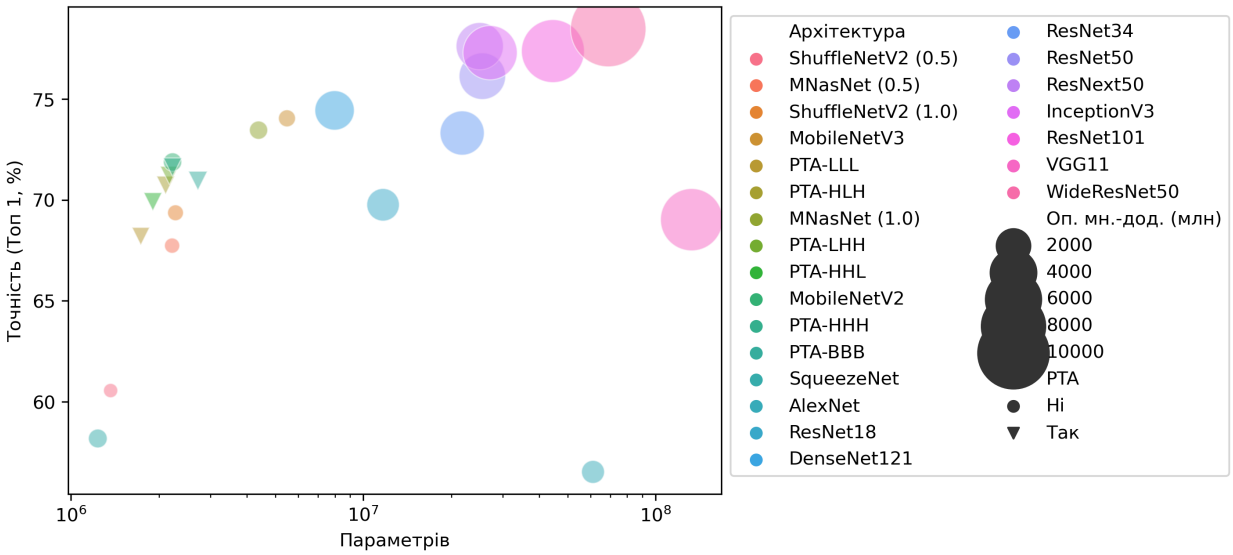


Рис. 2.2. Порівняння точності в залежності від кількості параметрів (за віссю X) та обчислювальної складності (розмір бульбашки)

стандартною практикою [32; 34–37; 40; 107; 135; 136; 155]. В роботі зерно дорівнює 17.

У таблиці 2.3 показано результати навчання мережі PTA у порівнянні з іншими провідними архітектурами нейронних мереж. Результати відсортовано за зменшенням ефективності (співвідношення точності до кількості операцій множення-додавання). Стандартної апаратної платформи для звітування часу виконання в літературі не представлено, саме тому для підрахунку ефективності використано кількість операцій множення-додавання, від якої напряму залежить час виконання. Конфігурації змінюваної мережі позначено як PTA-*, де * – конфігурація блоку PTA, застосована для звітування.

На рис. 2.2 представлено бульбашкову діаграму для результатів з таблиці 2.3. Показано точність (топ 1) для кожної з мереж в залежності від кількості параметрів (за віссю X) та обчислювальної складності (розмір бульбашки). Результати для мережі PTA виділено трикутниками.

З табл. 2.3 та рис. 2.2 можна побачити, що змінювана згорткова мережа в найлегшій конфігурації займає 5 місце з 17 розглянутих провідних

Таблиця 2.3. Якість та обчислювальна складність архітектур нейронних мереж на наборі даних ImageNet

Архітектура	Параметрів (млн)	Оп. мн.-дод. (млн)	Точність (Топ 1, %)	Точність (Топ 5, %)	Ефективність
ShuffleNetV2 (0.5)	1,36	40	60,552	81,746	1,513
MNasNet (0.5)	2,21	100	67,734	87,490	0,677
ShuffleNetV2 (1.0)	2,27	140	69,362	88,316	0,495
MobileNetV3	5,48	220	74,042	91,340	0,336
PTA-LLL	1,73	268	68,198	87,612	0,254
PTA-HLH	2,10	295	70,730	89,388	0,239
MNasNet (1.0)	4,38	310	73,456	91,510	0,236
PTA-LHH	2,17	308	71,210	89,712	0,231
PTA-HHL	1,90	303	69,910	88,798	0,230
MobileNetV2	2,22	318	71,878	90,286	0,226
PTA-HHH	2,22	318	71,624	89,954	0,225
PTA-BBB	2,71	368	70,948	89,462	0,192
SqueezeNet	1,23	350	58,178	80,624	0,166
AlexNet	61,10	710	56,522	79,066	0,079
ResNet18	11,68	1810	69,758	89,078	0,038
DenseNet121	7,97	2830	74,434	91,972	0,026
ResNet34	21,79	3660	73,314	91,420	0,020
ResNet50	25,55	4090	76,130	92,862	0,018
ResNext50	25,02	4230	77,618	93,698	0,018
InceptionV3	27,16	5710	77,294	93,450	0,013
ResNet101	44,54	7800	77,374	93,546	0,009
VGG11	32,86	7610	69,020	88,628	0,009
WideResNet50	68,88	11400	78,468	94,086	0,006

Таблиця 2.4. Порівняння часу виконання мереж PTA та MobileNetV2 для роздільної здатності 224×224

Тип пристрою	Процесор		Час виконання (мс)		
	Архіт.	Назва	MobileNetV2	PTA-LLL	PTA-BBB
Крайовий	RISC-V	Емулятор	2401,52±3,62	1949,20±3,42	2839,97±4,03
Бюдж. смартфон	ARM v7a	SD800	131,10±0,54	147,91±0,56	122,10±0,48
Прос. смартфон	AArch64	SD845	42,59±0,55	37,17±0,47	49,53±0,56
ЦП (Неттоп)	x86-64	i5-5200U	152,40±0,17	111,27±0,15	139,16±0,19
ЦП (Моб. ПК)	x86-64	i7-8750H	64,96±0,12	51,94±0,09	62,32±0,11
ГП (Моб. ПК)	Pascal	GTX 1050Ti	3,66±0,01	3,05±0,01	4,50±0,02
Відносно (%)			100,00	86,26	106,32

архітектур нейронних мереж за ефективністю та перевершує базову архітектуру MobileNetV2. Особливої уваги потребує той факт, що кожну модифікацію звичайної згорткової мережі будь-то за кількістю шарів (наприклад, ResNet18, ResNet34 із 18 та 34 шарами відповідно) або за множником ширини (ShuffleNetV2 (0.5) та (1.0)) необхідно навчати окремо. Напроти, всі варіанти змінюваної мережі PTA отримані після одного проходу навчання єдиної архітектури та наступної її зміни після навчання.

В таблиці 2.4 наведено заміри часу для роздільної здатності 224×224 на пристроях із різними обчислювальними можливостями, а саме: крайовий пристрій з архітектурою RISC-V, мобільні процесори бюджетного смартфону Qualcomm Snapdragon 800 (SD800) та просунутого смартфону Qualcomm Snapdragon 845 (SD845) на різних варіантах архітектури ARM (v7a та AArch64 відповідно), процесори неттопа та ноутбука Intel Core i5-5200U та Intel Core i7-8750H відповідно, а також на графічному процесорі Nvidia GTX 1050Ti. Заміри часу проводились за наступною процедурою: перші 100 замірів викидаються (так звана «фаза прогріву»), результати звітуються за наступними 1000 ітераціями із наведенням 95 % довірчого інтервалу. Для оцінки прискорення на інших пристроях, не наведених в табл. 2.4,

Таблиця 2.5. Результати перевірки гіпотези про статистичну значущість прискорення виконання мережі

Процесор	t-статистика	Ступенів свободи	p-значення
Емулятор	-177,86	1992	$<10^{-16}$
SD800	-24,19	1980	$<10^{-16}$
SD845	-14,56	1955	$<10^{-16}$
i5-5200U	-354,79	1936	$<10^{-16}$
i7-8750H	-170,80	1928	$<10^{-16}$
GTX 1050Ti	-76,58	1936	$<10^{-16}$

розраховано усереднений відносний час виконання. Для однакового вкладу кожної категорії пристроїв результати нормалізовано перед усередненням. Можна побачити, що на всіх пристроях мережа PTA дозволяє змінювати час виконання після завершення навчання, в середньому з 86,26 % до 106,32 % відносно мережі MobileNetV2.

Для даних з табл. 2.4 перевірено статистичну значущість t-критерієм Уелча із використанням бібліотеки SciPy [162]. t-критерій Уелча допускає різну дисперсію вибірок, яка враховується під час обчислення кількостей ступенів свободи. Нульовою гіпотезою H_0 є незначущість різниці середніх часу виконання архітектури PTA-LLL і архітектури MobileNetV2. Альтернативою нульової гіпотези є одностороння гіпотеза H_1, що середнє вибірки замірів часу архітектури PTA-LLL менше, ніж архітектури MobileNetV2. Результати є статистично значущими, як видно з в табл. 2.5, в якій наведено t-статистику, кількість ступенів свободи та p-значення вірогідності нульової гіпотези.

Далі проведено дослідження, наскільки вдалим є апріорний вибір частот конфігурацій PTA (табл. 2.2). Для підбору частот застосовано метод TPE (Three-Structured Parzen Estimators) [163] в реалізації бібліотеки Optuna [164]. Оскільки навчання мережі 200 епох за описаною вище конфігурацією займає більше 35 годин, а для підбору частот необхідно навчити мережу

велику кількість разів, тому процедуру навчання було спрощено: на кожній ітерації методу TPE нейронна мережа навчається 8 епох на зображеннях, зменшених до 156×156 та обрізаних до роздільної здатності 128×128. На валідаційній вибірці зображення оброблюються аналогічно. Методом TPE максимізується середня оцінка всіх конфігурацій PTA на валідаційному наборі. Всього проведено 80 ітерацій методу TPE, отримані оптимізовані частоти показано в табл. 2.6. Для порівняння там же наведені й апріорні

Таблиця 2.6. Апріорні частоти вибору конфігурацій PTA
та отримані за методом TPE

Конфігурація	Апріорні частоти	Частоти за методом TPE
PTA-BBB	0,000	0,411
PTA-HHH	0,450	0,270
PTA-LHH	0,150	0,216
PTA-HLH	0,150	0,012
PTA-HHL	0,150	0,008
PTA-LLL	0,100	0,083

частоти.

В табл. 2.7 наведено точність класифікації для змінюваної мережі при навчанні із апріорними та отриманими за методом TPE частотами вибору конфігурацій PTA. При цьому навчання проведено протягом 200 епох із роздільною здатністю навчальних зображень 208×208, як було описано на початку розділу. Підвищення точності за рахунок вибору оптимізованих частот відбулося лише на конфігурації PTA-BBB. Середня точність за апріорними частотами вибору PTA блоків вища, тому в подальших розділах завжди використовуються ці частоти.

Таблиця 2.7. Точність змінюваної згорткової мережі на наборі даних ImageNet при апріорних та отриманих за методом TPE частотах вибору конфігурацій PTA

Архітектура	Точність (%)			
	Апріорні частоти		Частоти за методом TPE	
	Топ 1	Топ 5	Топ 1	Топ 5
PTA-LLL	68,198	87,612	63,248	84,188
PTA-HHL	69,910	88,798	68,204	87,700
PTA-HLH	70,730	89,388	68,530	87,888
PTA-LHH	71,210	89,712	71,076	89,488
PTA-HHH	71,624	89,954	71,216	89,636
PTA-BBB	70,948	89,462	71,586	89,885
Усереднено	70,437	89,154	68,977	88,131

2.4 Експериментальні результати виконання змінюваної мережі на наборі даних антиспуфінгу CelebA-Spoof

В даному розділі навчання і тестування запропонованої архітектури нейронної мережі з адаптивними після навчання блоками здійснено для задачі антиспуфінгу обличчя.

Для навчання та оцінки мережі використовується набір даних CelebA-Spoof [56]. За проведеним аналізом, це найбільший доступний на сьогодні набір даних антиспуфінгу, що загалом містить 625 537 зображень (включаючи як приклади спуфінгу, так і достовірні фотографії) 10 177 людей. Фотографії зроблені з різним освітленням, умовами навколишнього середовища та різними камерами. Розглядаються кілька типів атак спуфінгу: друковані фотографії; фотографії, де вирізано обличчя; кадр із відео, представлений на планшеті чи телефоні; 3D-маска, коли друковане зображення накладається на обличчя людини. Окрім двійкової мітки зображення, що визначає чи є воно спуфінгом, набір даних містить багато інформації про тип підробки, умови

освітлення, середовище, а також мітки атрибутів обличчя (наявність посмішки, вусів, капелюха, окулярів тощо). У представленій в цій роботі мережі буде використано лише двійкову мітку, що визначає чи це спуфінг.

У відповідності до попередніх робіт [165; 166] для оцінки якості мережі використані наступні метрики:

— точність – це частка правильно класифікованих зображень до загальної кількості зображень;

— частота помилок класифікації представлення атак (APCER) – це частка зображень атак, неправильно класифікованих як звичайні зображення:

$$APCER = \frac{FP}{FP + TN}; \tag{2.4}$$

— частота помилок класифікації достовірного представлення (BPCER), тобто частка звичайних (достовірних) зображень, неправильно класифікованих як зображення атаки:

$$BPCER = \frac{FN}{FN + TP}; \tag{2.5}$$

— середня частота помилок класифікації (ACER) – це середнє значення APCER і BPCER:

$$ACER = \frac{APCER + BPCER}{2}, \tag{2.6}$$

де TP – істинно позитивний, тобто зображення – спуфінг, результат класифікації правильний; TN – істинно негативний, тобто зображення є достовірним, результат класифікації правильний; FP – хибно-позитивний: зображення достовірне, результат класифікації невірний; нарешті, FN – хибно-негативний: зображення спуфінг, результат класифікації невірний.

CelebA-Spoof визначає поділ на навчальну та тестову вибірки, а також кілька протоколів оцінювання. Далі результати наведено для протоколу “intra-test”, який використовується для загальної оцінки мережі. Навчальну множину було розділено на набори власне навчання та перевірки (валідації) у співвідношенні 80/20 із використанням псевдовипадкового генератора чисел із фіксованим зерном, рівним 17. Навчання проходить 20 епох, найкращі параметри мережі обираються за набором перевірки. Обчислення градієнта виконується на міні-пакетах із 32 зображень. В якості оптимізатора використовується метод адаптивного градієнтного спуску Adam [167] зі швидкістю навчання $\alpha = 10^{-4}$. На відміну від експериментів в розділі 2.3, навчання та тестування відбувається із повною точністю (fp32). Функція втрат перехресна ентропія. Результати показано на тестовому наборі.

Вхідні зображення попередньо обробляються: проводиться обрізка відповідно до рамок обличчя, які присутні для кожного зображення в наборі даних CelebA-Spoof. Далі зображення зменшується до роздільної здатності 128×128. На вхід мережі подаються кольорові (RGB) зображення. До тренувального набору застосовуються розширення зміни кольору (Color Jitter) та посилення ISO-шуму. Вхідні зображення нормалізуються. Ініціалізації ваг мережі проводиться псевдовипадковим генератором із фіксованим зерном.

Для порівняння було навчено 2 архітектури: оригінальну MobileNetV2 і запропоновану мережу PTA. Останню можна додатково налаштувати після навчання, тому в усіх наступних таблицях показано конфігурацію, для якої було виконано тестування.

У таблиці 2.8 показано порівняння якості на тестовому наборі. Якість навчання покращується із збільшенням точності і зменшенням показників APCER, BPCER, ACER. У кожному стовпчику найкращий результат показано червоним кольором, другий – синім. Як видно, конфігурації PTA домінують над оригінальною MobileNetV2 за всіма показниками. Легко бачити, що кон-

Таблиця 2.8. Порівняння якості оригінальної MobileNetV2 та запропонованої мережі PTA

Конфігурація	Точність (↑, %)	APCER (↓, %)	BPCER (↓, %)	ACER (↓, %)
MobileNetV2	96,74	1,07	4,18	2,63
PTA-HHH	96,89	**0,72**	4,12	2,42
PTA-LHH	96,84	**0,70**	4,20	2,45
PTA-HLH	97,04	0,80	3,88	2,34
PTA-HHL	**97,83**	2,30	**2,11**	**2,21**
PTA-LLL	**97,85**	2,53	**1,98**	2,26
PTA-BBB	97,49	1,21	3,05	**2,13**

Таблиця 2.9. Порівняння обчислювальної складності та часу виконання мереж PTA та MobileNetV2 для роздільної здатності 128×128

Конфігурація	К-сть парам. (↓, млн)	Оп. мн.-додав. (↓, млн)	Час SD845 (↓, мс)	Час SD800 (↓, мс)	Відносний час (↓)
MobileNetV2	2,23	104,15	21,32	94,23	1,00
PTA-HHH	2,23	104,15	21,27	94,12	1,00
PTA-LHH	2,17	100,63	**18,61**	**82,24**	**0,87**
PTA-HLH	2,11	**96,50**	19,67	86,96	0,92
PTA-HHL	**1,91**	99,00	19,27	85,15	0,90
PTA-LLL	**1,73**	**87,84**	**17,08**	**75,44**	**0,80**
PTA-BBB	2,73	120,47	22,72	100,47	1,07

фігурація PTA-HHH, еквівалентна за кількістю параметрів та операцій, також демонструє кращу якість за архітектуру MobileNetV2.

У табл. 2.9 представлено порівняння обчислювальної складності та часу виконання мереж PTA та MobileNetV2 для роздільної здатності 128×128. Наведено кількість параметрів у кожній із конфігурацій (у мільйонах). Для мережі PTA наведено кількість параметрів, що відповідають активній конфігурації. Далі оцінено кількість операцій множення-додавання (у мільйонах), виконаних під час прямого проходу мережі. Крім того, виміряно фактичний час виконання на мобільних пристроях на процесорах Snapdragon 845 і

Snapdragon 800 (SD845 і SD800 відповідно) в мілісекундах. Нарешті, показано відносне покращення часу виконання відносно MobileNetV2 (як виміряно на SD845). Конфігурація PTA-HHH має таку ж кількість параметрів і обчислень, що й MobileNetV2, і є еквівалентною з точки зору часу виконання на реальному пристрої. Конфігурація PTA-BBB використовує як легку, так і важку гілки у всіх трьох блоках PTA, тому потребує більше обчислень. Усі інші конфігурації, які використовують суміш важких і легких блоків, є швидшими, ніж MobileNetV2.

В табл. 2.10 наведено порівняння часу виконання мережі PTA та

Таблиця 2.10. Порівняння часу виконання мереж PTA та MobileNetV2 на пристроях із різними обчислювальними можливостями для роздільної здатності 128×128

Тип пристрою	Процесор		Час виконання (мс)		
	Архіт.	Назва	Без PTA	PTA-LLL	PTA-BBB
Крайовий	RISC-V	Емулятор	5580,20±4,24	4608,93±9,56	6585,54±4,54
Бюдж. смартфон	ARM v7a	SD800	94,23±0,14	75,44±0,13	100,47±0,14
Прос. смартфон	AArch64	SD845	21,32±0,10	17,08±0,10	22,72±0,10
ЦП (Неттоп)	x86-64	i5-5200U	48,09±0,10	36,55±0,08	44,01±0,08
ЦП (Моб. ПК)	x86-64	i7-8750H	20,46±0,09	15,80±0,08	20,56±0,08
ГП (Моб. ПК)	Pascal	GTX 1050Ti	3,81±0,02	3,17±0,02	4,53±0.03
Відносно (%)			100,00	79,86	107,02

MobileNetV2 на пристроях із різними обчислювальними можливостями: крайовий, смартфони (бюджетний та просунутий), центральні (мобільний ПК та неттоп) та графічний процесор ПК. Для кожного пристрою вказано архітектуру та назву процесора, на якому виконувалось тестування. Мережа PTA дозволяє змінювати час виконання від 79,96 % до 107,02 % в залежності від конфігурації відносно оригінальної MobileNetV2.

Оскільки під час навчання обираються по черзі важкі та легкі конфігурації PTA, загальний час навчання для мережі PTA зменшується, що експе-

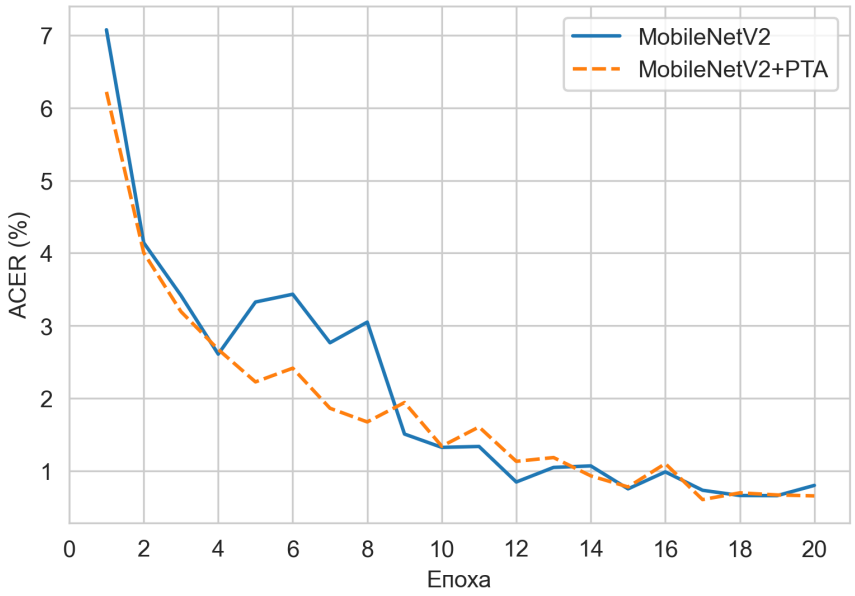


Рис. 2.3. ACER на перевірній виборці для MobileNetV2 проти змінюваної згорткової мережі

риментально підтверджується. У табл. 2.11 продемонстровано час навчання, найкращу точність та ACER, що було досягнуто кожною з мереж. Найкращий

Таблиця 2.11. Порівняння часу навчання, точності і ACER для MobileNetV2 проти змінюваної згорткової мережі

Конфігурація	Час епохи навч. (↓, хв.)	Загальний час навч. (↓, год.)	Краща точність (↑, %)	Кращий ACER (↓, %)
MobileNetV2	49,28	16,43	96,74	2,63
MobileNetV2+PTA	**43,11**	**14,37**	**97,85**	**2,13**

результат показано червоним. Час епохи показано в хвилинах, а загальний час навчання для 20 епох у годинах. Як видно, вибірка конфігурацій PTA під час навчання зменшує загальний час навчання на 14,34 %, одночасно покращуючи якість кінцевої мережі.

На рис. 2.3 показано ACER (нижчі значення кращі) протягом процедури навчання на перевірній виборці для оригінального MobileNetV2 (суцільна синя лінія) і змінюваної згорткової мережі (пунктирна помаранчева лінія).

Мережа PTA оцінюється на конфігурації PTA-HHH, яка еквівалентна з точки зору кількості параметрів і операцій множення-додавання оригінальному MobileNetV2. Як видно, частота помилок ACER зменшується швидше для змінюваної згорткової мережі, ніж для MobileNetV2.

Висновки до розділу 2

Для задачі класифікації розроблено змінювану згорткову нейронну мережу (мережу PTA) та метод її навчання, які дозволяють змінювати обчислювальну складність мережі під час або після навчання шляхом зміни кількості її згорткових шарів від 43 (для конфігурації з усіма легкими гілками) до 61 (для конфігурації, де у всіх блоках використовуються обидві гілки) для урахування обчислювальних можливостей пристроїв, на яких вона розгортається.

Проведено експерименти на наборі даних класифікації ImageNet, котрий, як правило, використовується в літературі для звітування результатів нових архітектур нейронних мереж та методів їх навчання. За результатами проведених експериментальних досліджень мережа PTA зайняла п’яте місце серед 17 провідних архітектур мереж за співвідношенням якість розпізнавання/час виконання. Зокрема час виконання розробленої мережі у порівнянні з оригінальною MobileNetV2 зменшено на 13,74 % при падінні точності (топ 1) на 3,68 %.

Проведено експерименти на наборі даних антиспуфінгу (задача класифікації) CelebA-Spoof, де мережа PTA перевершила оригінальну за всіма метриками та дозволила зменшити час виконання до 20 %. Зокрема, найкращі результати за метриками точності і BPCER має конфігурація PTA-LLL із значеннями (в дужках – результати MobileNetV2) у 97,85 % (проти 96,74 %) і 1,98 % (проти 4,18 %) відповідно; за APCER – PTA-LHH з 0,70 % (проти 1,07 %); за ACER – PTA-BBB з 2,13 % (проти 2,63 %). Загальний час навчан-

ня PTA моделі зменшено на 14,34 % порівняно із MobileNetV2.

Змінювана мережа PTA та метод її навчання допускають спільне використання з існуючими методами прискорення навчання та виконання нейронних мереж, такими як квантування та навчання зі змішаною точністю. На наборі даних ImageNet навчання проведено зі змішаною точністю (fp16), а на наборі CelebA-Spoof – із повною (fp32), та в обох випадках підтверджено ефективність мережі PTA. Використання змінюваної згорткової мережі з процедурою квантування є одним з можливих напрямків подальших досліджень.

РОЗДІЛ 3
СИСТЕМА КОНТРОЛЮ ДОСТУПУ ІЗ ВИКОРИСТАННЯМ ЗМІНЮВАНОЇ НЕЙРОННОЇ МЕРЕЖІ

Даний розділ присвячено розробці системи контролю доступу із використанням змінюваної згорткової нейронної мережі. Для цього розв'язано наступні задачі:

1. Застосовано представлену в попередньому розділі змінювану згорткову мережу та метод її навчання в підсистемі антиспуфінгу мобільної системи контролю доступу із RFID мітками, де аналіз зображень виконується на смартфоні користувача із метою зменшення вартості впровадження такої системи та підвищення її безпеки.
2. Розроблено реалізацію запропонованої системи контролю доступу із наступними застосунками: для смартфону (де виконується аналіз зображень та робота із RFID мітками), ПК (панель моніторингу та конфігурації) та серверної програми (Web API) для зберігання та обробки даних.

3.1 Запропонована мобільна система контролю доступу із RFID мітками та антиспуфінгом на основі змінюваної згорткової мережі

В розділі 1.3 було розглянуто існуючі системи контролю доступу та виявлено наступні недоліки: системи на основі RFID карток мають низьку безпеку, адже картку доступу може бути передано третій особі або втрачено, а фотографія людини, що входить в таких системах не перевіряється; біометричні системи потребують встановлення камер та коштовних систем розпізнавання облич, що створює суттєве навантаження на сервер.

Тому в роботі запропоновано нову схему контролю доступу, яка не вико-

ристовує сканер карток і має більш високі гарантії безпеки при вході через турнікет або розумні двері без використання камер відеоспостереження, для цього:

1. Сканери RFID-карток біля дверей замінено пасивними RFID-мітками, подібними до тих, які є в сучасних пластикових картках. У той час як сканер карток потребує підключення до комп'ютера, RFID-мітка – ні. Вона розміщується біля двері і зберігає її ідентифікатор та деяку додаткову інформацію. Мітки RFID можуть зберігати достатньо даних для запропонованої системи та набагато дешевшими за сканер RFID міток.

2. Для сканування мітки використовується смартфон користувача, а пластикова картка непотрібна. Якщо піднести смартфон до пасивної мітки, програма контролю доступу запуститься автоматично. Вся інформація про розташування дверей зчитується з мітки. Крім того, щоб уникнути необхідності встановлення окремої камери відеоспостереження, користувач повинен зробити фотографію на фронтальну камеру для підтвердження особи. Прочитана інформація з мітки разом із даними користувача надходить на сервер через корпоративну мережу Wi-Fi. Дані включають: інформацію про двері, ідентифікатор дверей, токен користувача, фотографію та результат її аналізу.

Блок-схему процедури відкриття дверей запропонованої системи контролю доступу представлено на рисунку 3.1, процедури аналізу зображення з камери на рисунку 3.2. Обидві процедури виконуються на смартфоні користувача. Остаточне рішення про надання доступу та відкриття дверей відбувається після обробки даних на сервері, як показано на рис. 3.3.

За запропонованою схемою в роботі розроблено систему контролю доступу, що складається з:

1. Додатку для смартфона, за допомогою якого адміністратори можуть відкривати та реєструвати двері або турнікети, а також встановлювати реєстраційні фотографії працівників компанії. Звичайні користувачі (не адміні-

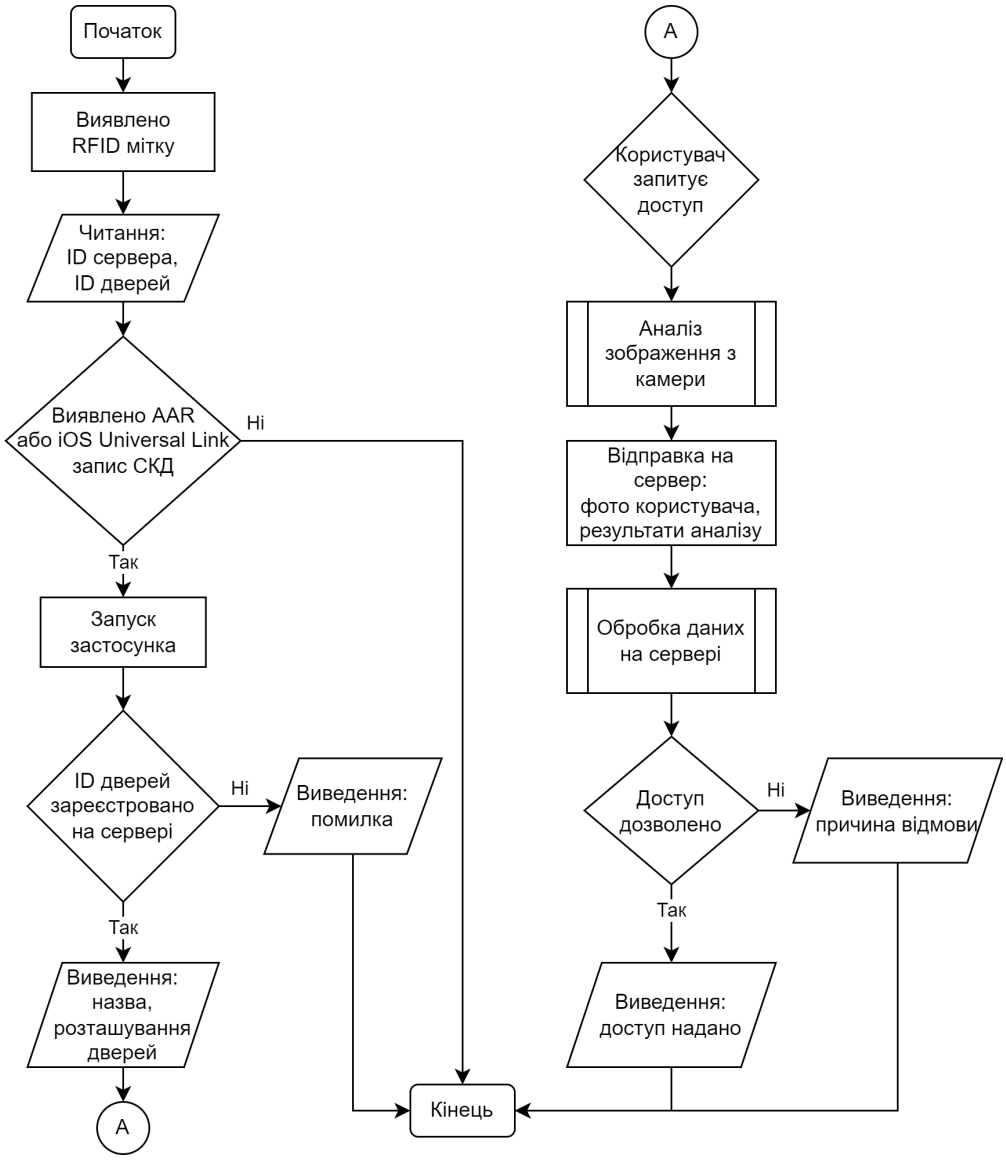


Рис. 3.1. Блок-схема процедури відкриття дверей запропонованої системи контролю доступу

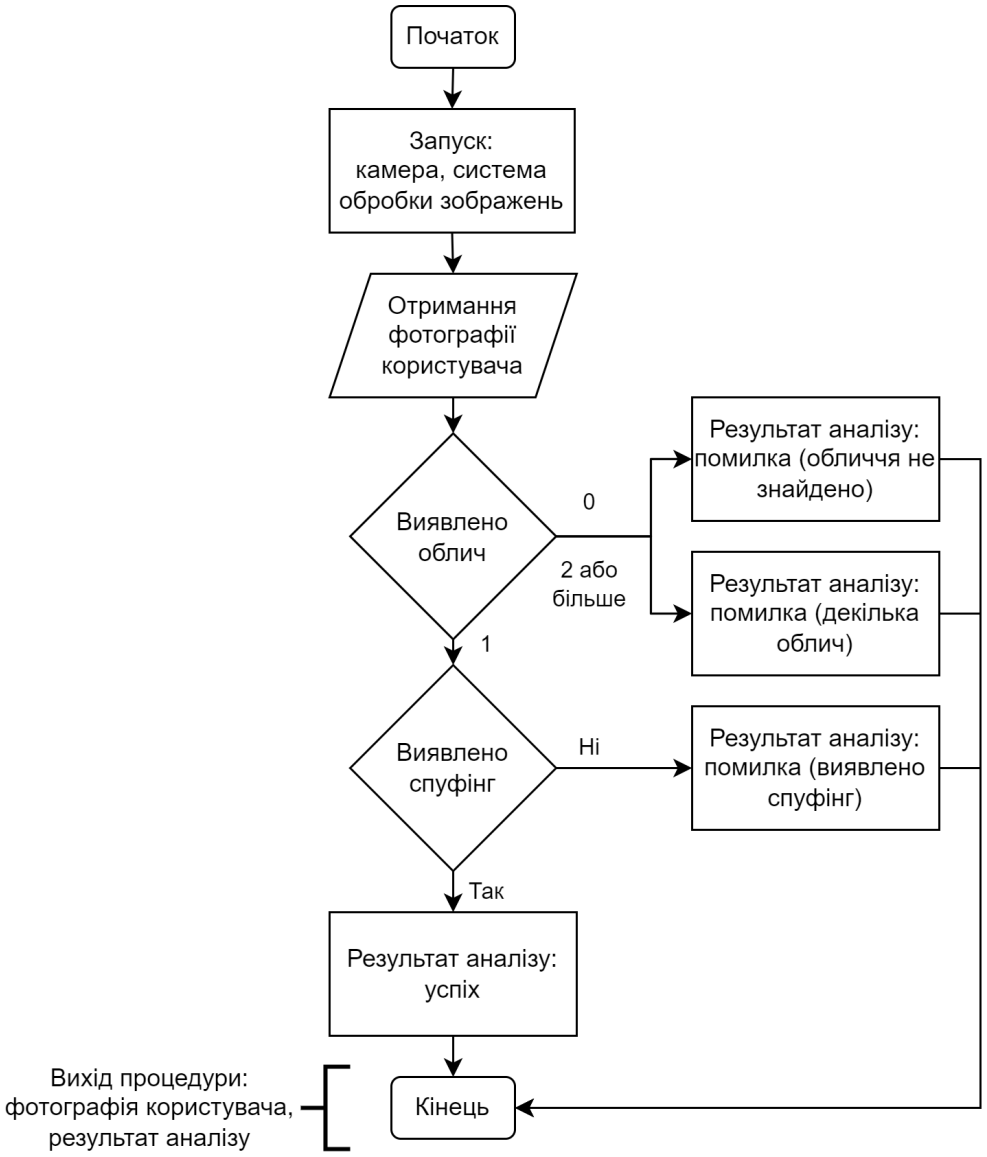


Рис. 3.2. Блок-схема процедури аналізу зображення з камери на мобільному пристрої

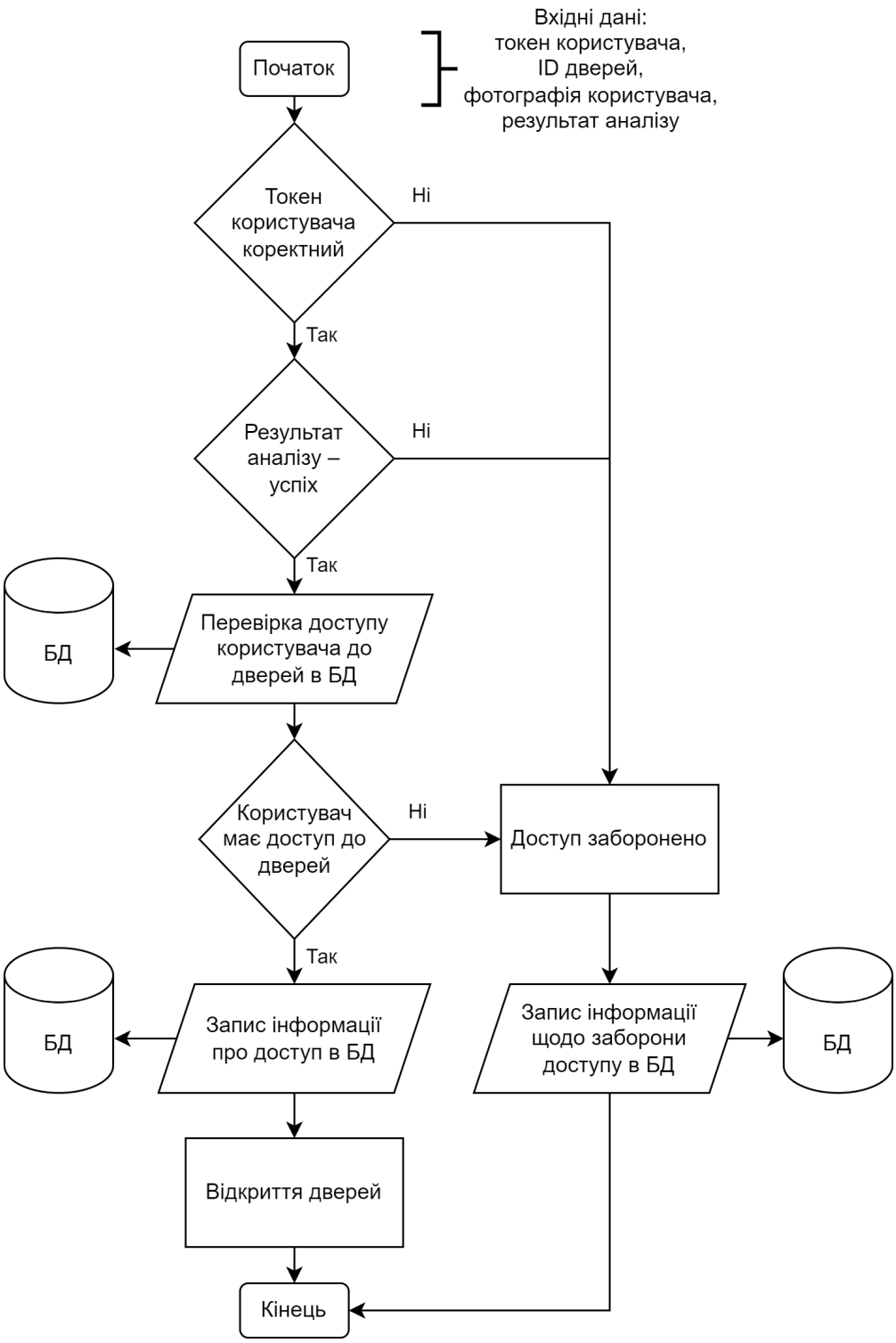


Рис. 3.3. Блок-схема процедури обробки даних на сервері для отримання доступу до дверей

стратори) використовують цю програму лише для відкриття дверей та турнікетів, інші функції недоступні.

2. Комп'ютерної програми моніторингу та конфігурації для адміністраторів компанії та служби безпеки, що надає інформацію про всі спроби відкриття дверей (як успішні, так і відхилені) та дозволяє керувати зареєстрованими користувачами та дверима.

3. Серверна програма (Web API), як оброблює, зберігає та надає дані для додатків на ПК і смартфоні.

Для розробки вищезазначених програм використано наступні технології і стандарти:

1. Стандартний прокол авторизації OAuth 2.0 [168], що визначає процедуру авторизації користувача та доступу до кінцевої точки Web API.
2. Мова програмування C# та ASP.NET Core [169] для розробки Web API серверної програми, що демонструють високу швидкість обробки даних та є широко використовуваними.
3. База даних із відкритим вихідним кодом PostgreSQL [170].
4. Для розгортання серверного компоненту використовується Docker при побудові контейнерів та Docker Compose [171] для автоматичної конфігурації Web API та бази даних. Розгортання системи можливе під Windows та Linux.
5. Мова програмування C# та фреймворк WinUI 3 [172] для розробки комп'ютерної програми моніторингу та конфігурації, що є останнім фреймворком для розробки застосунків для Windows від Microsoft.
6. Мова програмування C# та Xamarin.Forms [173] для підсистеми роботи із RFID мітками на смартфонах під ОС Android.
7. Мова програмування Kotlin, фреймворк Jetpack Compose [174] для розробки підсистеми аналізу зображення з камери на смартфоні. Для виконання нейронних мереж використовується libTorch [161].

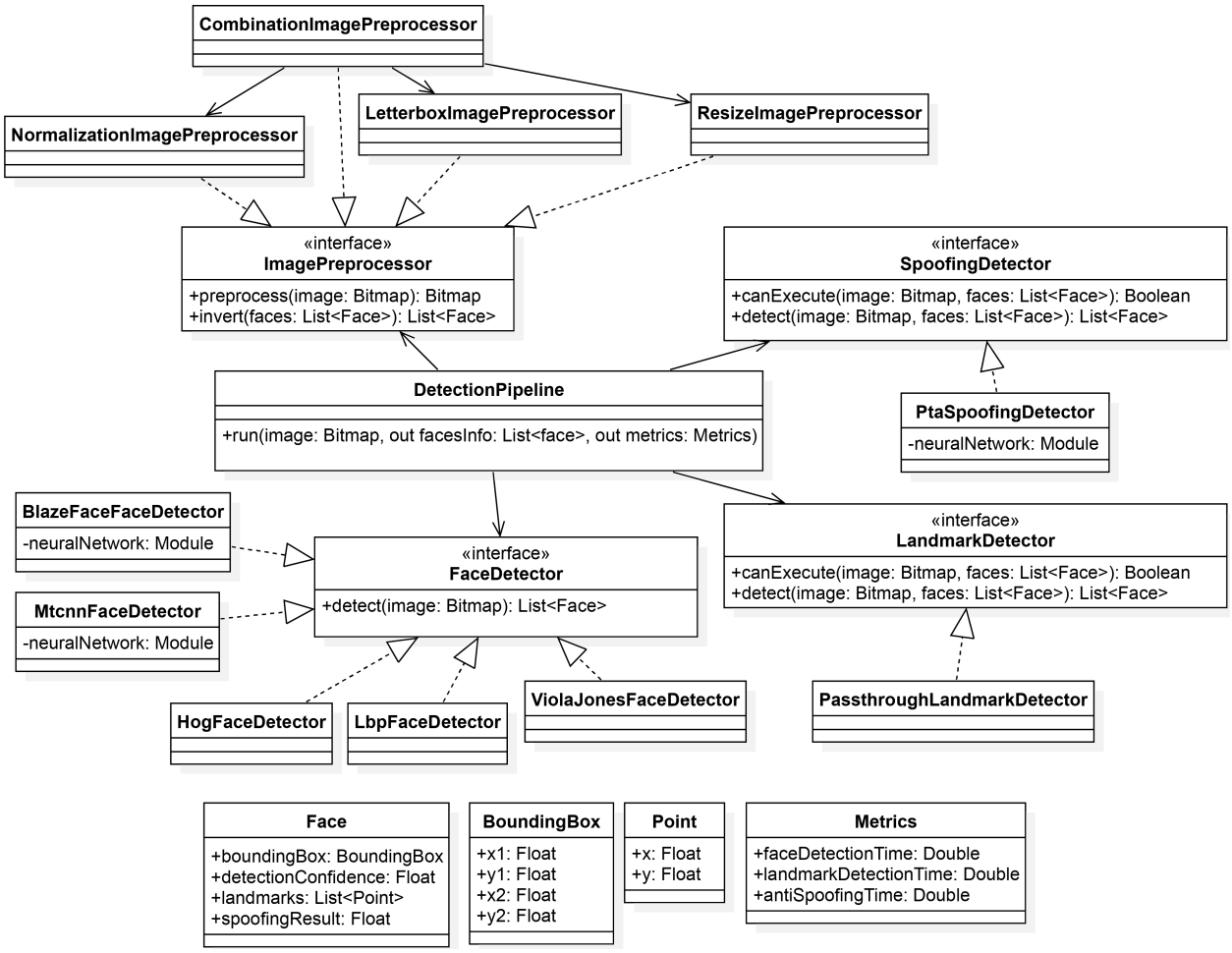


Рис. 3.4. UML-діаграма класів підсистеми аналізу зображення з камери на смартфоні

8. Для системи антиспуфінгу використовується змінювана нейронна мережа, запропонована в розділі 2 та навчена на наборі даних CelebA-Spoof, як описано в підрозділі 2.4.

На рис. 3.4 зображено UML-діаграму класів підсистеми аналізу зображення з камери на смартфоні. Зображення подається в конвеєр виявлення (DetectionPipeline), що проводить:

1. Масштабування та нормалізацію зображення (класи, що реалізують інтерфейс ImagePreprocessor).
2. Пошук облич через класи, що реалізують інтерфейс FaceDetector.
3. Пошук ключових точок через клас, що реалізує інтерфейс

LandmarkDetector. В даному випадку доступні лише ключові точки, отримані з методів спільного виявлення обличчя та ключових точок, таких як BlazeFace та MTCNN.

4. Виявлення спуфінгу за допомогою мережі PTA (реалізація інтерфейсу SpoofingDetector).

Результатом роботи конвеєру є інформація про обличчя (обмежувальна рамка, впевненість розпізнавання, перелік ключових точок, результат антиспуфінгу) та метрики часу виконання процедур виявлення.

3.2 Робота із RFID мітками

У запропонованій системі програмування міток проводиться адміністратором безпеки підприємства з використанням спеціального облікового запису за такою процедурою:

1. Заповнення даних про двері, до яких планується прикріпити мітку. Дані включають унікальну назву дверей та їх розташування.

2. У відповідь на запит реєстрації мітки, сервер генерує унікальний ідентифікатор для дверей. Щоб уникнути перезапису даних програмами сторонніх розробників або навмисного їх пошкодження, подальше програмування міток захищено паролем (можливість вбудована в обрані стандарти міток RFID). Пароль є глобальним для даної організації та надсилається із сервера.

3. Для завершення процесу реєстрації дверей адміністратор підносить пристрій до мітки на відстань 1-3 см для її програмування.

4. Нарешті, адміністратор налаштовує політики доступу користувачів для зареєстрованої мітки.

Для запису на мітку дані зберігаються в спеціальному двійковому форматі під назвою NDEF (NFC Data Exchange Format, формат обміну даними NFC). В Android цей двійковий формат реалізовано через спеціальний тип повідом-

Таблиця 3.1. Дані, що записуються на RFID-мітку при реєстрації дверей

Зміст	Розмір (байт)	Опис
ID сервера	16	GUID
ID дверей	4	Беззнакове ціле
Android Application Record	42	Залежить від довжини назви застосунку
iOS Universal Link	58	Залежить від довжини назви застосунку
Загалом	120	

лення NdefMessage, що містить записи NdefRecord [175]. В системі на мітку записується така інформація:

1. Глобальний унікальний ідентифікатор сервера (GUID сервера), який використовується для перевірки належності користувача до організації. Слід зазначити, що відмінною рисою GUID є висока випадковість генерації, тобто колізії (генерації одного і того самого GUID) фактично неможливі [176].

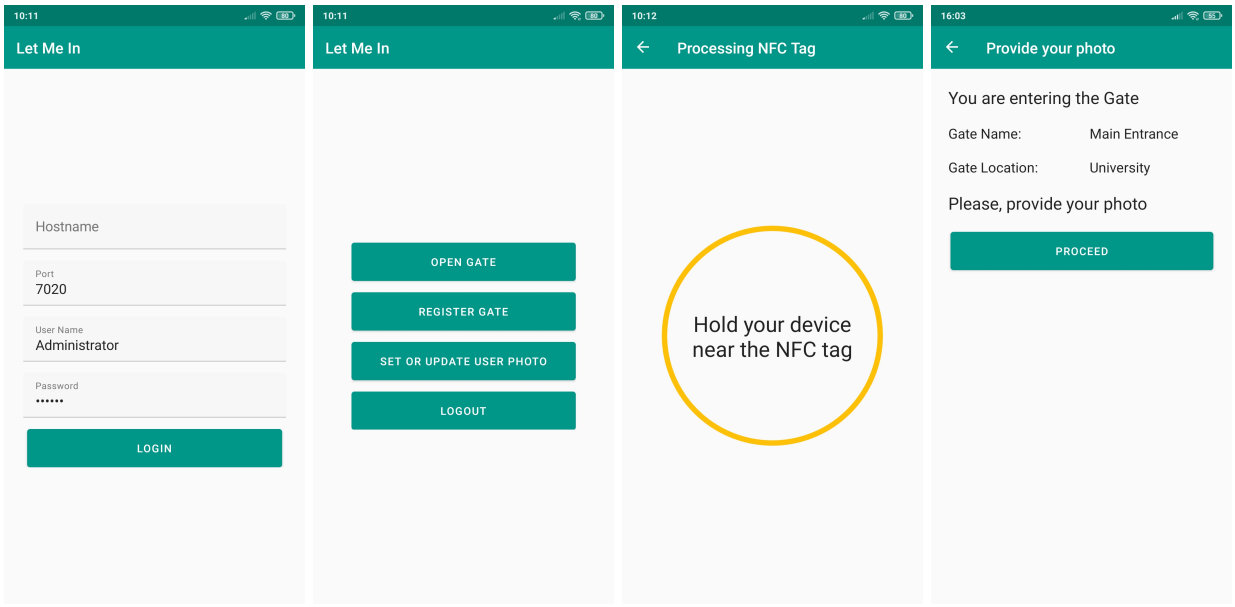
2. Беззнакове ціле число, що представляє ідентифікатор дверей.

3. Спеціальний запис програми Android (Android Application Record, AAR) [175], який використовується для миттєвого запуску програми, коли пристрій підноситься до мітки, єдина вимога – пристрій має бути розблоковано.

4. Аналогічний запис для пристроїв iOS, що містить так зване універсальне посилання (Universal Link) [177].

Розрахунок розміру кожного поля показано в табл. 3.1. Загалом на мітку записуються 120 байт, а отже, кожен з розглянутих в підрозділі 1.3.1 стандартів RFID-міток має достатньо пам’яті для розробленої системи.

Відмітимо, що смартфони Apple довгий час не містили чип NFC [67]. Навіть після його появи використання NFC було обмежено лише функціями Apple Pay. Наразі API NFC швидко додаються до операційної системи Apple iOS. З iOS 11.0 стало можливим зчитувати мітки RFID, а в iOS 13.0 – записувати [178]. Нові пристрої також мають підтримку фонового читання мі-



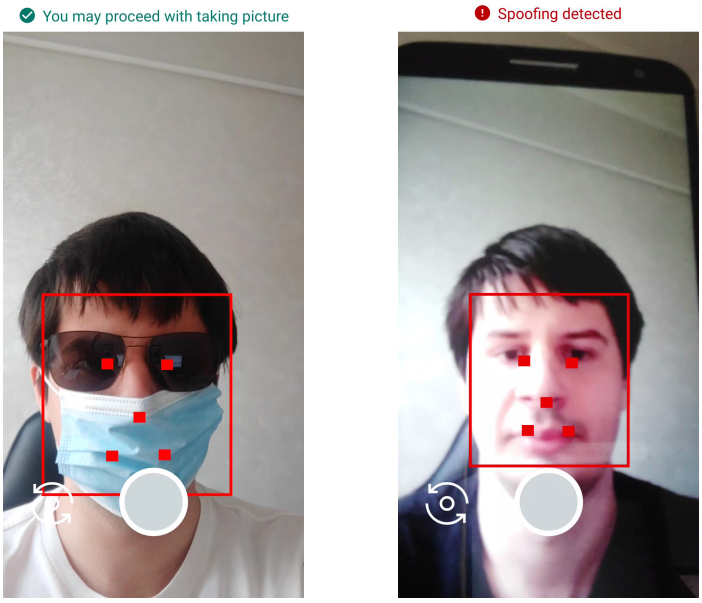
(а) Вхід (б) Вибір дії (в) Обробка мітки (г) Інформація про двері

Рис. 3.5. Скріншоти основних етапів роботи програми розробленої системи контролю доступу для смартфонів

ток [179] – функція, здебільшого аналогічна AAR, але на відміну від Android, додаток не запускається автоматично, а користувачеві видається сповіщення, яке запрошує запустити його. Отже, всі необхідні для розробленої системи функції доступні на обох мобільних платформах.

3.3 Опис користувацького інтерфейсу та приклади роботи програми

Приклади роботи програми для смартфона показано на рис. 3.5. Щоб увійти (рис. 3.5а), користувач має надати інформацію для підключення до сервера (ім’я хоста або IP-адреса, порт) та інформацію про користувача (ім’я користувача та пароль). Система зберігає надану інформацію, доки користувач вручну не вийде із системи. На цьому етапі налаштування програми завершено. Далі користувач може вибрати одну з таких дій: відкрити або зареєструвати двері, встановити або оновити фотографію користувача, вийти



(а) Виявлення обличчя

(б) Антиспуфінг

Рис. 3.6. Перевірка фотографії системами виявлення обличчя та антиспуфінгу на мобільному пристрої користувача

з системи (рис. 3.5б). Якщо користувач не має адміністративного дозволу, доступним є відкриття дверей та вихід. Щоб відкрити або зареєструвати двері, користувачеві потрібно піднести свій пристрій до мітки (рис. 3.5в). Зауважимо, що для відкриття дверей необов'язково відкривати застосунок, а достатньо розблокувати смартфон. Під час реєстрації дверей адміністратор повинен надати інформацію про назву та місцезнаходження дверей – потім ця інформація надається при відкритті дверей (рис. 3.5г).

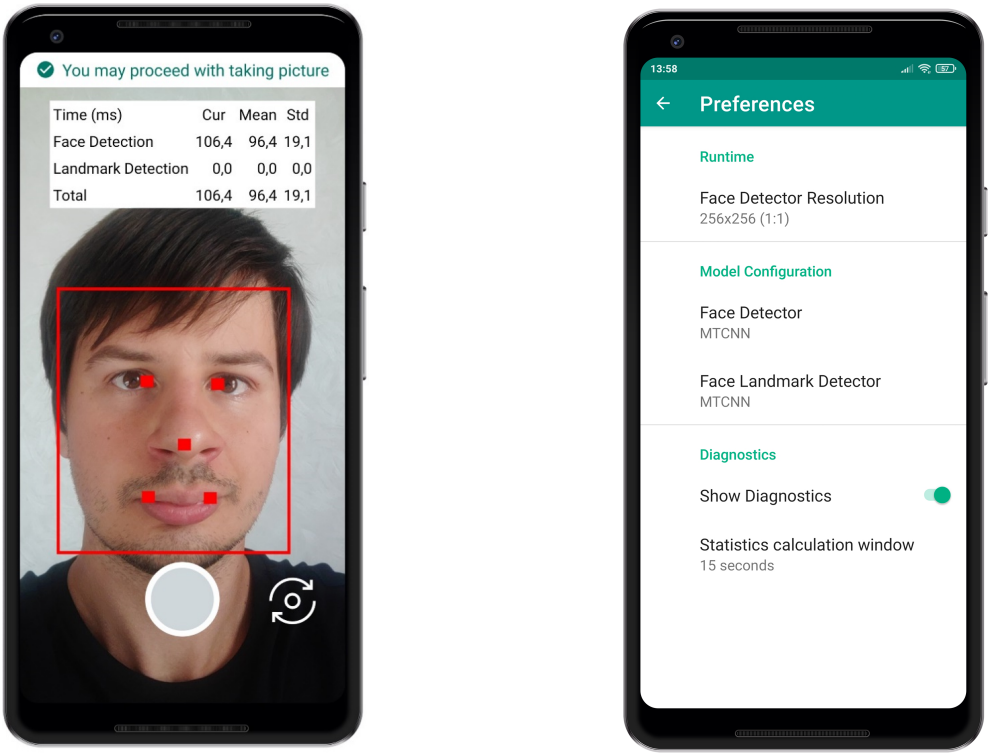
Щоб відкрити двері, користувачеві необхідно надати свою фотографію. На ній здійснюються пошук обличчя мережею та перевірка системою антиспуфінгу. На рис. 3.6а представлено приклад виявлення обличчя та ключових точок. Носіння маски або окулярів не створює проблем для системи, і користувачеві дозволено продовжити, як видно з рядка стану зверху. Для пошуку облич використовується мережа MTCNN. Для додаткової безпеки завжди працює підсистема антиспуфінгу, щоб виявити випадки, коли користувач на-

магається увійти із чиєюсь фотографією. Зазначена підсистема заснована на змінюваній згортковій мережі, яку було запропоновано в розділі 2 та навчено на наборі даних CelebA-Spoof, як описано в підрозділі 2.4.

На рис. 3.6б користувач намагається відкрити двері за допомогою відео, яке відтворюється на іншому смартфоні – доступ заборонено. Реалізація системи розпізнавання облич для порівняння рис обличчя між реєстраційною фотографією та знімком, зробленим під час відкриття дверей, не є задачею даної роботи, але за потреби й таку систему можна впровадити, наприклад, за допомогою методу FaceNet [86]. В такому разі, якщо обличчя належать різним людям, доступ також буде відхилено.

Для адміністратора системи доступні налаштування методу пошуку облич та роздільної здатності вхідного зображення, для заданої конфігурації розраховується середній час виконання та стандартне відхилення (рис. 3.7). Для роздільної здатності доступні як типові значення: 128×128, 256×256, 480×360, 640×480; так і наднизькі: 32×32, 64×64 – що дозволяє проаналізувати, чи є метод здатним до роботи із зображеннями з низьким рівнем деталізації. Наступним пунктом є конфігурація застосунку, де можна обрати метод пошуку облич (метод Віоли-Джонса [29], LBP [180], HOG [30], MTCNN [69], BlazeFace [38]), та чи необхідно отримувати з нього інформацію про ключові точки (якщо це підтримується методом). Методи реалізовано за допомогою бібліотек OpenCV [181], Dlib [182], libTorch [161]. Далі доступна конфігурація діагностики, яка відображається під час виконання методу (див. таблицю на рис. 3.7а). Діагностика може відображатися у вигляді середнього та вибіркового стандартного відхилення, розрахованого за заданий проміжок часу, або ж у вигляді масиву спостережень із окремими значеннями часу роботи методів.

Для вищезазначених методів в табл. 3.2 показано результати вимірів часу для зображень із роздільною здатністю 256×256. μ – середнє, sd – вибіркове



(а) Діагностична інформація (б) Панель конфігурації

Рис. 3.7. Конфігурація застосунку

Таблиця 3.2. Порівняння часу виконання методів на мобільних пристроях

Метод	Snapdragon 845		Snapdragon 800	
	μ (мс)	sd	μ (мс)	sd
MTCNN	98,9	24,1	210,3	100,4
BlazeFace	83,0	18,8	253,2	57,4
LBP	14,5	4,2	72,8	19,0
HAAR	71,2	7,2	401,8	87,6
HOG	25,2	4,0	90,6	28,1

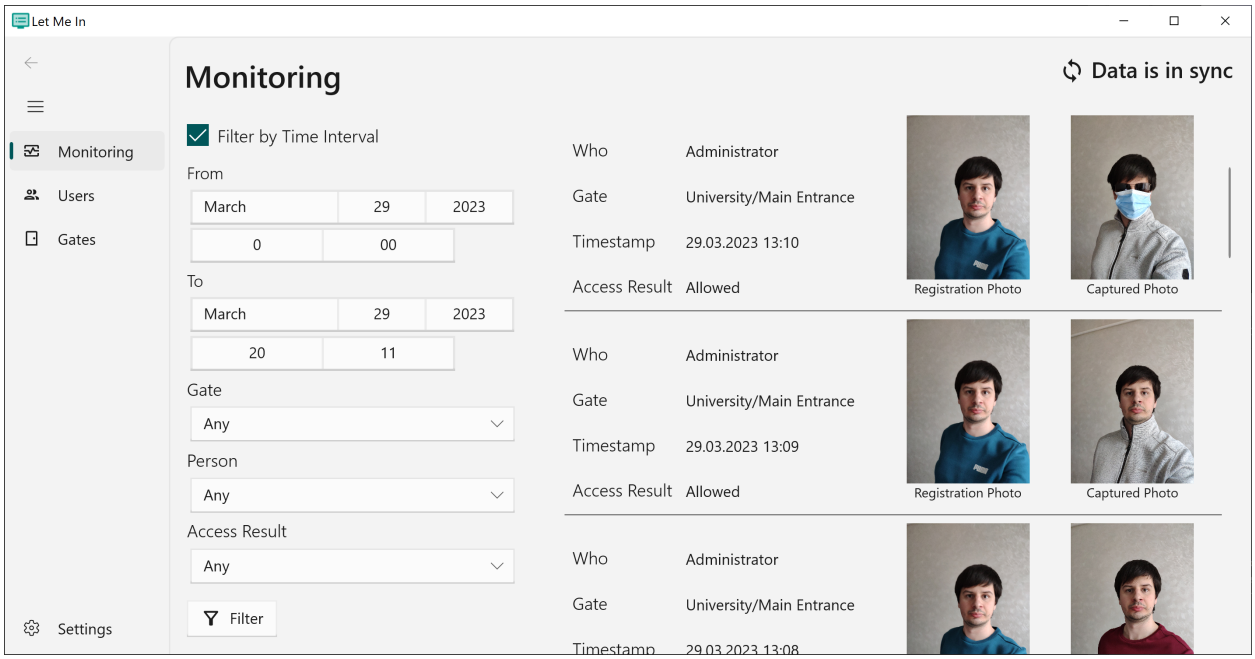


Рис. 3.8. Вкладка моніторингу розробленої системи контролю доступу

стандартне відхилення. Статистика обчислюється за проміжок у 30 секунд, однаковий час відводиться на сцени без облич, з одним великим обличчям і з 2 маленькими обличчями, що є типовими сценаріями для розробленої системи контролю доступу. Використовується модифікація BlazeFace для задньої камери. Методи виконуються у 2,13–5,64 рази повільніше на бюджетному процесорі Snapdragon 800, ніж на просунутому Snapdragon 845. Варто зазначити, що методи на основі нейронної мережі менше сповільнились, ніж «класичні». Це призвело до того, що каскади Хаара почали виконуватись довше, ніж MTCNN і BlazeFace, тоді як на Snapdragon 845 каскади Хаара були швидшими.

За результатами аналізу джерел [14; 29; 30; 38; 69; 180], методом пошуку облич за замовчуванням обрано MTCNN, оскільки він має високу швидкість роботи та якість пошуку.

Основним інструментом моніторингу є програма для комп’ютерів. Для доступу до неї потрібен обліковий запис адміністратора. Головна панель – це панель моніторингу (рис. 3.8). Перелік спроб доступу показано праворуч,

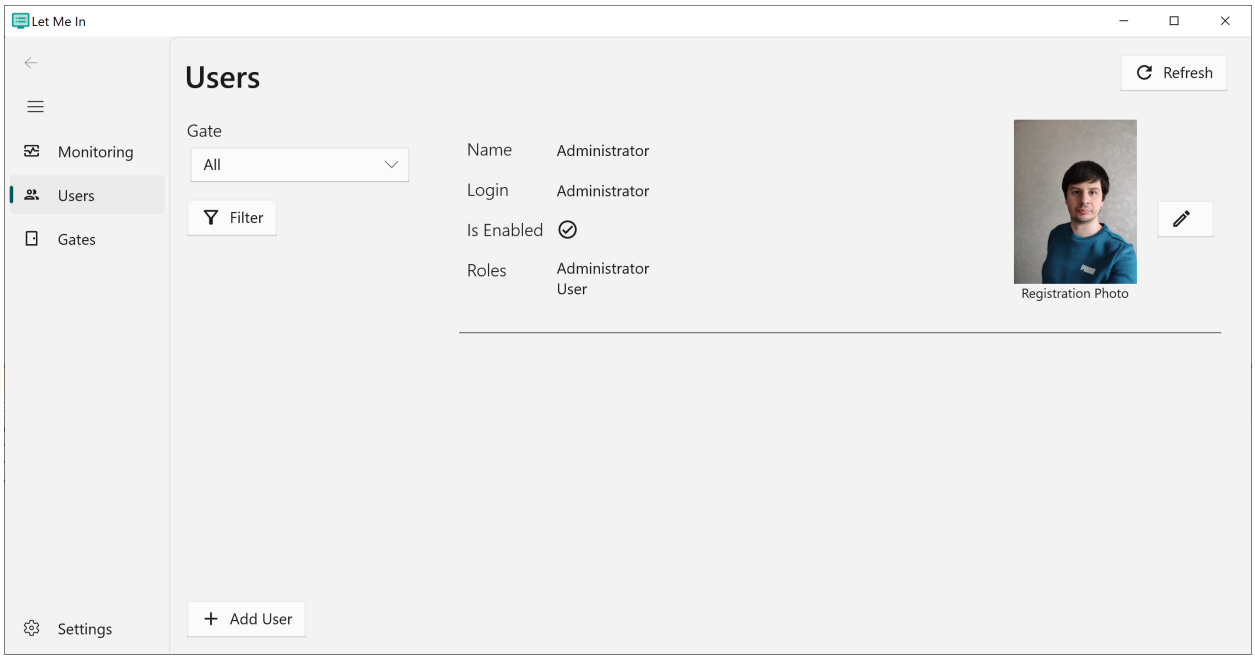


Рис. 3.9. Панель конфігурації користувачів

останні доступи – зверху. Система автоматично синхронізує дані. Для кожної події представлена така інформація: ім’я користувача, реєстраційна фотографія та фотографія, зроблена біля дверей, розташування та ім’я дверей, позначка часу доступу та результат (дозволено чи заборонено). За замовчуванням відображаються всі доступи. За потреби на панелі зліва їх можна відфільтрувати: за часовим фільтром (наприклад, у робочий час або поза ним); дверима, до яких було зроблено спробу доступу; особою чи результатом доступу (дозволено, заборонено). Кожен із фільтрів можна залишити порожнім.

Також програму можна використовувати для додавання нових або редагування зареєстрованих користувачів (рис. 3.9). Доступною для редагування є така інформація: ім’я користувача, логін, чи активний користувач, ролі (користувач, адміністратор), а також список дверей, які користувачеві дозволено відкривати. Щоб встановити або оновити фотографію користувача, слід використовувати додаток для смартфона, як описано раніше. Користувачів, які мають доступ до певних воріт, можна відфільтрувати за допомогою спадного списку ліворуч.

Насамкінець, інформацію про двері (назву дверей, розташування та чи вони активовані) можна переглядати та редагувати на однойменній вкладці. Щоб зареєструвати нові дверей, використовується додаток на смартфоні, оскільки потрібен фізичний доступ для програмування RFID-мітки, прикріпленої до дверей.

Разом із вищеописаними частинами, призначеними для користувача, розроблено REST API, який можна використовувати для розширення функціональних можливостей системи на вимогу компанії. Крім того, очікується, що API буде використовуватися для інтеграції зі сторонніми системами розумних замків або турнікетами і дозволить достатньо просто впровадити розроблену систему в наявній інфраструктурі.

Висновки до розділу 3

В даному розділі:

1. Представлено нову мобільну систему контролю доступу із RFID мітками і підсистемою антиспуфінгу, розробленою на основі змінюваних згорткових мереж, яка дозволяє зменшити навантаження на сервер та підвищити захищеність самої системи контролю доступу. Запропонована система контролю доступу включає:
   — адміністративну панель для налаштування політик доступу до підприємства;
   — систему моніторингу з фільтрами за часом доступу, користувачем та контрольованими дверми з RFID-мітками;
   — мобільний додаток, що здійснює пошук облич та здійснює перевірку зображення на спуфінг. Додаток створений для реєстрації і відмикання контрольованих дверей;
   — серверну програму, яка оброблює, зберігає та надає дані для дода-

тків на ПК і смартфоні.

Впровадження розробленої системи дозволяє знизити вартість систем контролю доступу за рахунок заміни стаціонарного RFID-сканера на дешеву мітку, а також відмовитися від встановлення камер відеоспостереження, оскільки користувач робить фотографію на свій мобільний телефон, коли відмикає двері, а його фотографія перевіряється системою антиспуфінгу.

2. Розроблено мобільний застосунок, який опрацьовує вхідний сигнал з камери в реальному часі за допомогою змінюваної згорткової нейронної мережі прямо на мобільному пристрої. Розроблений застосунок в подальшому можна гнучко налаштувати до роботи із будь-якими задачами пошуку об’єктів та класифікації зображень для наукових досліджень, на підприємствах, для кінцевих користувачів із метою зменшення навантаження на центральний сервер підприємства та для пришвидшення отримання результатів, в тому числі в умовах відсутнього або повільного доступу до мережі Інтернет.

РОЗДІЛ 4
ЗМІНЮВАНА ЗГОРТКОВА МЕРЕЖА ДЛЯ ЗАДАЧІ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ ТА МЕТОД ЇЇ НАВЧАННЯ

Даний розділ присвячено розробці архітектури змінюваної нейронної мережі та методу навчання такої мережі в задачі сегментації зображень. Для цього вирішені наступні задачі:

1. Розроблений в розділі 2 блок змінюваної згорткової мережі впроваджено в архітектуру нейронної мережі U-Net, що дозволило обирати конфігурацію для виконання при розгортанні на пристроях із різною обчислювальною потужністю навіть після того, як процедуру навчання закінчено.
2. Здійснено програмну реалізацію запропонованого методу для задачі сегментації зображень на наборі даних CamVid, проведено експериментальні дослідження, результати яких підтвердили ефективність запропонованого методу.

4.1 Постановка задачі сегментації

Нехай X – вхідне зображення, яке представлено у формі 3-вимірного тензора розміром $W \times H \times C$, де W, H, C – ширина, висота та кількість каналів (кольорів) зображення відповідно. Зазвичай використовуються кольорові зображення з 3 каналами, по одному для червоного, зеленого та синього кольорів. Тоді задача сегментації зображень [26; 34; 183–185] полягає в тому, щоб знайти таку функцію $\Phi : X \to Q$, яка за вхідним зображенням X надає матрицю $\hat{Q}$, розміру $W \times H$, де $\hat{Q}_{i,j} \in \{1 \dots K\}$ – це номер класу (типу об'єкта), до якого належить піксель (i, j) вхідного зображення. K – кількість класів (типів об'єктів) на зображенні. Загалом, задача сегментації є задачею попі-

ксельної класифікації зображення, на виході отримуємо маску для кожного типу об'єктів. Часто зустрічається формулювання проблеми, коли результатом функції Ф є тензор $\hat{Y}$ розміру $W \times H \times K$ із розподілом ймовірностей належності пікселя до певного класу, далі $\hat{Q}_{i,j} = \text{argmax}_k \hat{Y}_{i,j,k}$. Для навчання функції Ф використовуються анотовані набори даних із відомою розміткою зображення на класи $Q_{i,j}$, тоді $Y_{i,j}$ – істинний унітарний вектор, що формується за формулою $\forall k \neq Q_{i,j} : Y_{i,j,k} = 0; y_{i,j,Q_{i,j}} = 1$. Перелік класів (типів об'єктів) є кінцевим та фіксованим. Оскільки на зображенні може зустрітися об'єкт поза межами переліку, такі об'єкти позначаються як «нейтральний» клас або клас «фону». Якщо на зображенні існує декілька об'єктів одного типу, то кожен екземпляр об'єкту отримає один і той самий номер класу. В такій постановці задача називається «семантичною сегментацією». Також існують: «сегментація екземплярів», яка виокремлює маски екземплярів деякого одного типу об'єкту; «паноптична сегментація», що поєднує семантичну та сегментацію екземплярів, і кожен об'єкт отримує як номер класу до якого він належить, так і номер екземпляру об'єкту на зображенні. Найчастіше використовується саме семантична сегментація, тому на ній і зосереджено подальше дослідження.

В рамках одного набору даних для навчання та тестування нейронної мережі використовуються різні зображення для оцінки узагальнюючої можливості мережі. Для оцінки якості зазвичай обчислюються метрики $\text{Dice}_{\text{score}}$ або коефіцієнт Жаккара. Найпоширенішими наборами даних для сегментації зображень є CamVid [26], PascalVOC [184], Cityscapes [185].

4.2 Архітектура змінюваної мережі U-Net+PTA

Багато практичних задач виграють від швидкої та точної сегментації зображень. Найкращу точність розв'язання поставленої задачі показують згор-

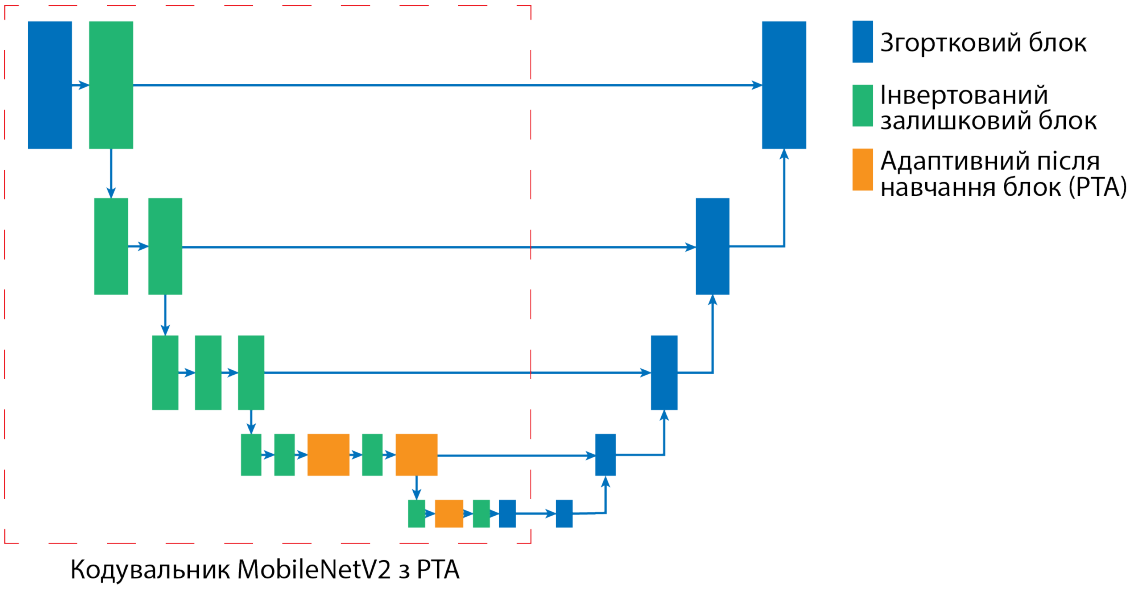


Рис. 4.1. Запропонована архітектура U-Net+PTA для сегментації зображення

ткові нейронні мережі. Застосунки включають обробку медичних [28] та супутникових [27] зображень, автономне водіння [26] тощо. Типові архітектури нейронних мереж для сегментації зображень необхідно повністю визначити до початку процедури навчання, а для зміни архітектури необхідні додаткові етапи навчання. Це є обмеженням, оскільки мережа може працювати не лише на потужному сервері, а й на мобільному чи крайовому пристрої [20; 186]. Навчання окремих мереж для кожної категорії пристроїв є досить неефективним. В ідеалі зміна конфігурації мережі повинна виконуватися динамічно під час виконання.

У розділі 2 було представлено змінювану згорткову мережу для задачі класифікації. У цьому розділі запропоновано мережу для задачі сегментації зображень. Базовою архітектурою обрано U-Net з кодувальником MobileNetV2. Аби побудована нейронна мережа була змінюваної, до мережі додано 3 адаптивні блоки PTA, як показано на рис. 4.1. Блоки PTA використовуються лише в кодувальнику MobileNetV2, декодувальник залишається незмінним. На схемі згорткові блоки відображаються синім кольором, інвертовані залишкові – зеленим, адаптивні після навчання (PTA) – помаранчевим. Шляхом вбудо-

ви PTA блоків кількість згорткових шарів мережі можна змінювати під час або після навчання від 54 (для конфігурації з усіма легкими гілками) до 72 (для конфігурації, де у всіх блоках використовуються обидві гілки). Для порівняння оригінальна мережа U-Net має 63 згорткових шари, кількість або характеристики яких не може бути змінено після навчання.

Для навчання мережі обрано функцію втрат $\text{Dice}_{\text{loss}}$, що широко використовується для задачі сегментації, а для вимірювання якості роботи мережі – функцію $\text{Dice}_{\text{score}}$ [187]:

$$Dice_{loss} = 1 - \sum_{i=1}^{W}\sum_{j=1}^{H}\frac{2\sum_{k=1}^{K}\hat{Y}_{i,j,k}Y_{i,j,k} + \epsilon}{\sum_{k=1}^{K}\hat{Y}_{i,j,k}^2 + \sum_{k=1}^{K}Y_{i,j,k}^2 + \epsilon}; \tag{4.1}$$

$$Dice_{score} = \sum_{i=1}^{W}\sum_{j=1}^{H}\frac{2\sum_{k=1}^{K}\hat{Y}_{i,j,k}Y_{i,j,k} + \epsilon}{\sum_{k=1}^{K}\hat{Y}_{i,j,k}^2 + \sum_{k=1}^{K}Y_{i,j,k}^2 + \epsilon}; \tag{4.2}$$

де W, H – ширина, висота зображення відповідно, K – кількість класів, які слід розрізняти, $\hat{Y}_{i,j}$ – розподіл ймовірностей на виході мережі, $Y_{i,j}$ – істинний унітарний вектор, ϵ – мала константа.

4.3 Експериментальні результати роботи мережі сегментації U-Net+PTA

Для навчання та оцінки мережі використовувався широко відомий набір даних CamVid [26], що містить фотографії вулиць з автомобіля розміром 480×360 пікселів. Набір даних поділено на підмножини навчання (367 зображень), перевірки (101 зображення) і тестування (233 зображення). Для всіх підмножин доступні маски сегментації однакового 480×360 розміру. Завдання полягає в тому, щоб навчити мережу сегментувати частини зображення на один із наступних класів: небо, будівля, стовп, дорога, тротуар, дерево, дорожній знак, паркан, автомобіль, пішохід, велосипедист, або «не розмічено» – для елементів інших класів.

Таблиця 4.1. Порівняння $\text{Dice}_{\text{score}}$ для мережі U-Net та змінюваної згорткової мережі U-Net+PTA для задачі сегментації на наборі даних CamVid

Конфігурація	$\text{Dice}_{\text{score}}$ (↑)
U-Net	0,8583
U-Net+PTA-HHH	0,8666
U-Net+PTA-LHH	0,8659
U-Net+PTA-HLH	**0,8670**
U-Net+PTA-HHL	0,8660
U-Net+PTA-LLL	0,8647
U-Net+PTA-BBB	**0,8667**

Для навчального набору використовуються зображення, масштабовані до розміру 256×256. Для зберігання оригінального співвідношення ширини та висоти застосовано техніку каше. Застосовано розширення зміни кольору (Color Jitter) та випадкового кадрування (Random Crop). Мережі U-Net і U-Net+PTA навчено протягом 600 епох, починаючи з випадкової ініціалізації. Попереднє навчання нейронної мережі не виконувалося. Розмір міні-пакета встановлено 8. Adam [167] зі швидкістю навчання $\alpha = 10^{-3}$ використано як оптимізатор. Результати показані на тестовому наборі. Для навчання та тестування мережі використано відеокарту NVIDIA GTX 1050Ti. Навчання відбувається із фіксованим зерном (дорівнює 17) псевдовипадкового генератора чисел для відтворюваності результатів.

У табл. 4.1 на тестовому наборі показано $\text{Dice}_{\text{score}}$ для архітектури U-Net (кодувальник MobileNetV2) та змінюваної згорткової мережі U-Net+PTA (кодувальник змінювана мережа PTA), позначеної як «U-Net+PTA-*», де * – конфігурація блоку PTA, застосована для виконання. Найкращий результат показано червоним кольором, другий – синім. Як добре видно, всі конфігурації на основі PTA показують вищу якість, ніж оригінальний U-Net. Найкращі результати має PTA-HLH, потім PTA-BBB. Зауважимо, що всі конфігурації PTA були отримані з єдиної архітектури, навченої лише один раз. Завдяки змінюю-

Таблиця 4.2. Порівняння складності мереж архітектур U-Net і U-Net+PTA

Конфігурація	К-сть парам. (млн)	Оп. мн.-додав. (млн)
U-Net	6,63	871,80
U-Net+PTA-HHH	6,63	871,80
U-Net+PTA-LHH	6,58	868,58
U-Net+PTA-HLH	6,51	**864,16**
U-Net+PTA-HHL	**6,31**	866,65
U-Net+PTA-LLL	**6,14**	**855,49**
U-Net+PTA-BBB	7,12	888,12

ваній архітектурі конфігурацію можна вибрати після завершення навчання. За рахунок навчання із вибором конфігурацій блоку PTA для PTA-HHH спостерігається покращення якості у порівнянні з еквівалентною архітектурою U-Net.

Як видно з табл. 4.1, всі конфігурації PTA демонструють вищу якість у порівнянні з U-Net без PTA. Найкращий результат досягнуто PTA-HLH (поліпшення $Dice_{score}$ 0,0087), потім PTA-BBB (+ 0,0084). Конфігурація PTA-HHH є кращою за оригінальну мережу на 0,0083 за зазначеним показником, що свідчить про ефективність методу навчання мережі із вибором конфігурацій PTA блоку.

В таблиці 4.2 показано порівняння складності мережі U-Net та запропонованої U-Net+PTA. Найкращий результат показано червоним кольором, другий – синім. Відображено таку інформацію: конфігурація блоків мережі, кількість параметрів мережі, кількість операцій множення та додавання.

В таблиці 4.3 показано порівняння часу виконання мереж U-Net та U-Net+PTA на пристроях із різними обчислювальними можливостями: крайовий, смартфони (бюджетний – Qualcomm Snapdragon 800 та просунутий – Qualcomm Snapdragon 845), центральні процесори (неттоп – Intel Core i5-

Таблиця 4.3. Порівняння часу виконання мереж U-Net та U-Net+PTA на пристроях із різними обчислювальними можливостями

Тип пристрою	Розмір зображ.	Процесор	Час виконання (мс)		
			U-Net	PTA-LLL	PTA-BBB
Крайовий	64×64	Емулятор	5732,17±5,29	4788,33±6,75	6686,46±5,41
Бюдж. смарт.	128×128	SD800	195,28±1,31	185,23±1,13	196,67±1,14
Прос. смарт.	128×128	SD845	61,15±0,12	60,16±0,12	62,92±0,15
ЦП (Неттоп)	128×128	i5-5200U	31,71±0,18	30,16±0,17	32,25±0,16
ЦП (Моб. ПК)	128×128	i7-8750H	28,57±0,10	27,07±0,08	29,45±0,10
ГП	128×128	GTX 1050Ti	10,17±0,02	9,85±0,01	10,47±0,01
Відносно (%)			100,00	93,91	104,66

5200U та мобільний ПК – Intel Core i7-8750H) та графічний процесор ПК (NVIDIA GTX 1050Ti). Для кожного пристрою вказано архітектуру та назву процесора, на якому виконувалось тестування. Для забезпечення точних і відтворюваних вимірювань заміри часу усереднюються за 1000 спробами та надається 95 % довірчий інтервал. Як видно, мережі із увімкненою легкою гілкою скоріші за оригінальну мережу U-Net на 6,09 %.

У табл. 4.4 показано загальний час навчання та найкращий результат $\text{Dice}_{\text{score}}$ для мереж U-Net та U-Net+PTA. Найкращий результат $\text{Dice}_{\text{score}}$

Таблиця 4.4. Час навчання та порівняння найкращого результату для мереж U-Net і U-Net+PTA

Конфігурація	Загальний час навчання (↓, хв.)	Кращий $\text{Dice}_{\text{score}}$ (↑)
U-Net	161,6	0,8583
U-Net+PTA	**158,6**	**0,8670**

вибрано з усіх можливих конфігурацій PTA за один прохід навчання. Обидві мережі навчались 600 епох. Як видно, вищий показник $\text{Dice}_{\text{score}}$ для мережі PTA досягнено без збільшення часу навчання.

Відмітимо, що переваги використання 3 блоків PTA в мережі U-Net для за-

дачі сегментації дещо менші (прискорення 6,09 %), ніж у випадку мережі PTA для задачі класифікації (прискорення 20,14 %). Це можна пояснити тим, що загалом U-Net є суттєво більшою мережею за кількістю параметрів та операцій множення-додавання, ніж звичайна MobileNetV2. В табл. 4.5 показано по-

Таблиця 4.5. Кількість параметрів мереж та операцій множення-додавання для виконання задач класифікації та сегментації

Архітектура	Задача	К-сть парам. (млн)	Оп. мн.-додав. (млн)
MobileNetV2	Класиф.	2,23	104,15
PTA-LLL	Класиф.	1,73	87,84
U-Net	Сегм.	6,63	871,80
U-Net+PTA-LLL	Сегм.	6,14	855,49

рівняння кількості параметрів мережі та операцій множення-додавання для виконання задачі класифікації (класиф.) і сегментації (сегм.). В подальшому має сенс дослідити можливість збільшення кількості PTA блоків в змінюваній згортковій мережі U-Net+PTA.

Висновки до розділу 4

Запропоновані в другому розділі блоки PTA інтегровано в мережу U-Net, яка використовується для задачі сегментації зображень. Шляхом вбудови блоку PTA кількість згорткових шарів можна змінювати від 54 (для конфігурації з усіма легкими гілками) до 72 (для конфігурації, де у всіх блоках використовуються обидві гілки). Отриману змінювану згоркову мережу навчено на наборі даних CamVid, де продемонстровано зменшення часу виконання мережі на 6,09 % у порівнянні з оригінальною мережею U-Net.

Мережу розгорнуто на крайовому, мобільних, персональних комп'ютерах та графічному процесорі. Показано, що остаточну навчену мережу PTA можна перемикати під час виконання між шістьма конфігураціями, котрі від-

різняються обчислювальною складністю та якістю. Важливо, що всі конфігурації мають вищу якість, ніж оригінальна мережа U-Net (із $Dice_{score} = 0,8583$). Конфігурація PTA-LLL є найшвидшою та, в середньому за усіма пристроями, досягає прискорення на 6,09 % ($Dice_{score} = 0,8647$).

РОЗДІЛ 5
МЕТОД Λ-ШАБЛОНІВ ПРИСКОРЕННЯ НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ЗА КІЛЬКОМА ПРИКЛАДАМИ

В попередніх розділах використовувались мережі PTA та U-Net+PTA, котрі необхідно навчати на великому наборі даних. Навчати можна з випадковою ініціалізацією ваг мережі або з отриманими в результаті трансферного навчання. Однак, навчальний набір даних при цьому залишається достатньо великим. Для навчання за кількома прикладами на клас (1, 5, 15 тощо) необхідно використовувати інші архітектури нейронних мереж та методи мета-навчання. Так, якщо мережа PTA має змінну кількість згорткових шарів – від 43 до 61, а U-Net+PTA – від 54 до 72, то для оптимізаційного мета-навчання найчастіше використовується мережа CNN4 з лише 4 згортковими шарами, як показано в розділі 1.5. Причинами використання суттєво менш глибокої мережі є: 1) замала кількість прикладів для навчання мережі із великою кількістю шарів та параметрів (таких як ResNet, MobileNetV2 або мережі PTA); 2) велика обчислювальна складність процедури адаптації мережі до нових класів.

Очевидно, що навіть один блок PTA із змінною кількістю згорткових шарів (від 3 до 9) є завеликим для вбудови в мережу CNN4 та не зможе її прискорити. Тим не менш, методи навчання за кількома прикладами є важливими та потребують прискорення процедури адаптації до нових класів, а отже, для них необхідна розробка іншого методу пришвидшення нейронної мережі, фокусом якого буде саме прискорення процедури адаптації. Для цього розв’язані наступні задачі:

1. Розроблено метод Λ-шаблонів прискорення оптимізаційного мета-

навчання, що за рахунок зміни складності нейронної мережі дозволяє значно пришвидшити адаптацію до нових класів задач за мінімальних втрат якості. Крім того, запропонований метод підвищує якість однокрокової адаптації мета-навчання.

2. Розроблено застосунок для навчання нейронної мережі для задачі класифікації зображень за малою кількістю навчальних прикладів із розробленим методом Λ-шаблонів оптимізаційного мета-навчання.
3. Проведено експериментальне дослідження запропонованого методу на наборі даних CIFAR-FS та підтверджено його ефективність.

5.1 Постановка задачі навчання за кількома прикладами

Мета-навчання є підходом, що дозволяє «навчитися навчатися». Його метою є навчити нейронну мережу $\Phi(\theta, \cdot)$ так, щоб вона була здатна адаптуватися до нових раніше невідомих задач по невеликій кількості прикладів. Процедура навчання визначається через концепцію задач, які вибираються з усього простору задач $\rho(\mathcal{T})$ предметної області. Навчальна задача визначається кортежем $\mathcal{T} = \{S, Q\}$ з набором підтримки $S = \{X_S, y_S\}$ і набором запитів $Q = \{X_Q, y_Q\}$ [137; 144; 147; 149]. X_S – приклади, y_S – правильна розмітка для навчання мережі. X_Q, y_Q – це вхідні дані та правильна розмітка, яка використовується для оцінки мережі. Набір підтримки S є малим та використовується для адаптації (або навчання) мережі до нової задачі. Кількість прикладів на клас в наборі S дорівнює K і позначається далі як «K-прикл.». Величина K зазвичай знаходиться в діапазоні від 1 до 20, хоча жорстку верхню межу не визначено. Кількість класів N, які мережа повинна розрізняти між собою, позначається як «N-клас.».

Процедура оптимізаційного мета-навчання складається з наступних двох кроків:

1. Крок адаптації, який обчислює ваги мережі у формі функції $\theta'(\theta)$, які мінімізують специфічну для задачі $\mathcal{T}$ помилку. Для класифікації використовується, як правило, функція втрат крос-ентропія:

$$\mathcal{L}(y, \Phi(\theta, X)) = -\sum_i y_i \log \Phi(\theta, X_i). \tag{5.1}$$

2. Обчислення мета-градієнта, який оновлює мета-параметри θ.

Ідея такої процедури навчання полягає в тому, що, знайшовши хороші вагові коефіцієнти θ, можна буде адаптуватися до нових раніше небачених задач за невеликою кількістю навчальних прикладів.

Для подальших модифікацій в даній роботі обрано метод MAML оскільки:

1. Він може бути застосований до багатьох різних задач, зокрема: класифікації зображення, навчання з підкріпленням, навіть для анімації облич (як описано в [20]) тощо. Фокусом даної роботи є класифікація зображень.
2. Метод MAML є основою інших підходів оптимізаційного мета-навчання, а отже, запропоновані далі модифікації можуть бути в подальшому застосовані й до них.

Ключовою задачею, яка вирішується за допомогою запропонованого далі підходу, є прискорення фази адаптації MAML до нових задач.

5.2 Ідея методу Λ-шаблонів для мета-навчання за кількома прикладами

Метод MAML передбачає початковий випадковий вибір навчальних задач $\mathcal{T}_i, i = 1, \ldots, H$ з простору всіх задач $\rho(\mathcal{T})$ навчального набору даних. Першим його етапом є адаптація до нових класів задач, яка полягає в мінімізації функції втрат (5.1) для кожної з задач на наборі підтримки, виконуючи кілька

кроків стохастичного градієнтного спуску. Метод ітеративно настроює вагові коефіцієнти моделі як функцію $\theta^{(i,j)}(\theta)$:

$$\theta^{(i,j)} = \theta^{(i,j-1)} - \alpha \nabla_{\theta^{(i,j-1)}} \mathcal{L}\left(y_{S_i}, \Phi\left(\theta^{(i,j-1)}, X_{S_i}\right)\right), \tag{5.2}$$

де $\alpha > 0$ – крок адаптації, i – номер задачі, якій відповідають ваги, j – ітерація методу. Зауважимо, що $\theta^{(i,0)}(\theta) \equiv \theta$, ініціалізується випадково.

Далі, використовуючи параметри мережі $\theta^{(i,j)}$, побудовані для кожної з H задач, метод оновлює мета-параметри θ:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{Q_i \in \mathcal{T}_i} \mathcal{L}\left(y_{Q_i}, \Phi\left(\theta^{(i,P)}, X_{Q_i}\right)\right), \tag{5.3}$$

де $\beta > 0$ – швидкість навчання, P – кількість кроків адаптації.

Відмітимо, що в (5.2) ваги $\theta^{(i,j)}$ обчислюються на наборі підтримки S, а втрати в (5.3) – на наборі запитів Q. Крім того, в (5.2) використовується градієнт функції втрат за коефіцієнтами конкретної задачі $\theta^{(i,j-1)}$, а в (5.3) враховується її градієнт за мета-параметрами θ.

На відміну від попередніх досліджень, фокусом даної роботи є зменшення часу адаптації мережі до нових класів. Очікується, що після навчання мережу може бути адаптовано до багатьох різних задач (наприклад, через онлайн платформу), а отже, прискорення адаптації є важливим.

Введемо поняття Λ-шаблону. Для згорткової нейронної мережі, яка має B шарів, шаблон адаптації визначається як $\Lambda = \{\Lambda_1, \Lambda_2, \ldots, \Lambda_B\}$, де Λ_l – індикативна функція, що вказує шари мережі, параметри яких слід оновити під час зворотного поширення:

$$\forall l : \Lambda_l = \begin{cases} 1, & \text{якщо параметри шару } l \text{ оновлюється,} \\ 0, & \text{інакше.} \end{cases} \tag{5.4}$$

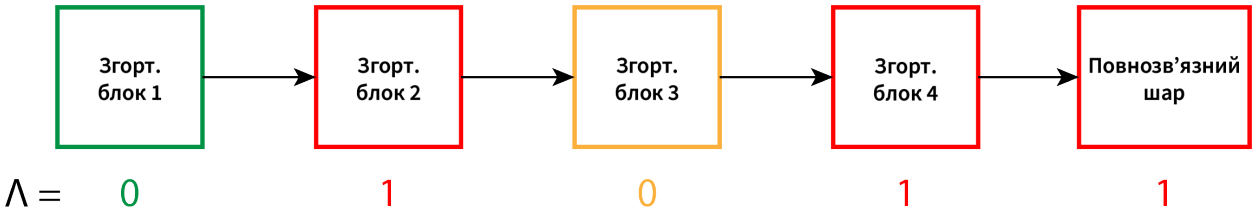


Рис. 5.1. Схема методу зворотного поширення для шаблону Λ

Пропуск оновлення параметрів мережі для певних шарів дозволяє в цих випадках не обчислювати градієнт для них, а отже прискорити фазу адаптації. На рис. 5.1 показано схему методу зворотного поширення для шаблону $\Lambda = \{0, 1, 0, 1, 1\}$ для згорткової мережі, яка містить $B = 5$ шарів, з них 4 згорткових і один повнозв'язний шар. Така архітектура береться як приклад та може бути довільною. Зворотне поширення виконується в напрямку, протилежному до стрілок. Червоним кольором позначено шари, в яких градієнти обчислюються, ваги мереж оновлюються; жовтим – градієнти обчислюються, параметри не оновлюються; зеленим – пропускаються як обчислення градієнта, так і оновлення ваг мережі.

Для розглянутого шаблону $\Lambda = \{0, 1, 0, 1, 1\}$:

1. У згортковому блоці 4 та повнозв'язному шарі обчислюється градієнт і оновлюються ваги.
2. У згортковому блоці 3 вагові коефіцієнти не оновлюються, але градієнти обчислюються, оскільки згортковий блок 2 потребує оновлення вагових коефіцієнтів.
3. У згортковому блоці 1 обчислення градієнтів або оновлення параметрів не виконується.

Шаблон називається повним, якщо $\forall l : \Lambda_l = 1$, у цьому випадку запропонована фаза адаптації еквівалентна представленій в MAML. В методі розглядаються всі можливі шаблони Λ, за винятком $\forall l : \Lambda_l = 0$, коли жодні параметри не можуть бути оновлені, а отже і адаптація неможлива.

Деталізуємо запис θ, виділивши в ньому матриці ваг кожного шару мережі: $\theta = \{\theta_1, \ldots, \theta_B\}$, тоді оновлена формула адаптації (замість (5.2)) застосовується до кожного шару $l = 1, \ldots, B$ та виглядає наступним чином:

$$\theta_l^{(i,j)} = \theta_l^{(i,j-1)} - \mathbf{\Lambda}_l \alpha \nabla_{\theta_l^{(i,j-1)}} \mathcal{L}\left(y_{S_i}, \Phi\left(\theta^{(i,j-1)}, X_{S_i}\right)\right). \tag{5.5}$$

Отже, одну ітерацію процедури навчання складають наступні кроки:

1. З простору задач $\rho(\mathcal{T})$ випадково обрати H задач $\mathcal{T}_i, i = 1, \ldots, H$.
2. Для кожної задачі $\mathcal{T}_i = \{S_i, Q_i\}$, де $S_i = \{X_{S_i}, y_{S_i}\}, Q_i = \{X_{Q_i}, y_{Q_i}\}$:
3. Для кожного кроку адаптації $j = 1, \ldots, P$:
4. Для кожного шару мережі $l = B, \ldots, 1$:
5. Адаптувати мережу за формулою (5.5), використовуючи S_i.
6. Оновити мета-параметри θ за формулою (5.3), використовуючи набір Q_i та специфічні для кожної з задач ваги $\theta^{(i,P)}, i = 1, \ldots, H$.

5.2.1 Конфігурація мережі та методу навчання для проведення експериментів

Для проведення експериментів в даній роботі було повторно реалізовано метод MAML відповідно до статті [137] та змінено процедуру адаптації мережі так, аби вона оновлювала лише ваги, визначені шаблоном Λ, відповідно до (5.4) та (5.5).

Автори MAML визначили архітектуру згорткової нейронної мережі для експериментів на miniImageNet, у літературі цю мережу зазвичай називають «CNN4». Її буде використано і в даній роботі. Вона має 4 згорткові блоки, за якими йде повнозв'язний шар. Кожен із згорткових блоків складається із згорткового шару із розміром ядра 3 та доповненням 1, за яким іде пакетна нормалізація [41], активація ReLU та субдискретизація максимуму із розміром ядра 2. У кожному згортковому шарі 32 фільтри. Кількість виходів у пов-

нозв'язному шарі визначається числом N для N-клас. задачі. Навчання виконується за допомогою методу градієнтного спуску Adam [167] зі швидкістю навчання $\beta = 10^{-3}$ і кроком адаптації $\alpha = 0,01$. Навчання відбувається протягом 600 епох. Далі завжди використовуються $H = 4$ задачі для оновлення θ. Кожна епоха має 100 випадково обраних задач $\mathcal{T}_i$. При навчанні для оновлення градієнта використовується $K \cdot N$ зразків для K-прикл. N-клас. та 15 зразків на клас для оцінки, дотримуючись [147].

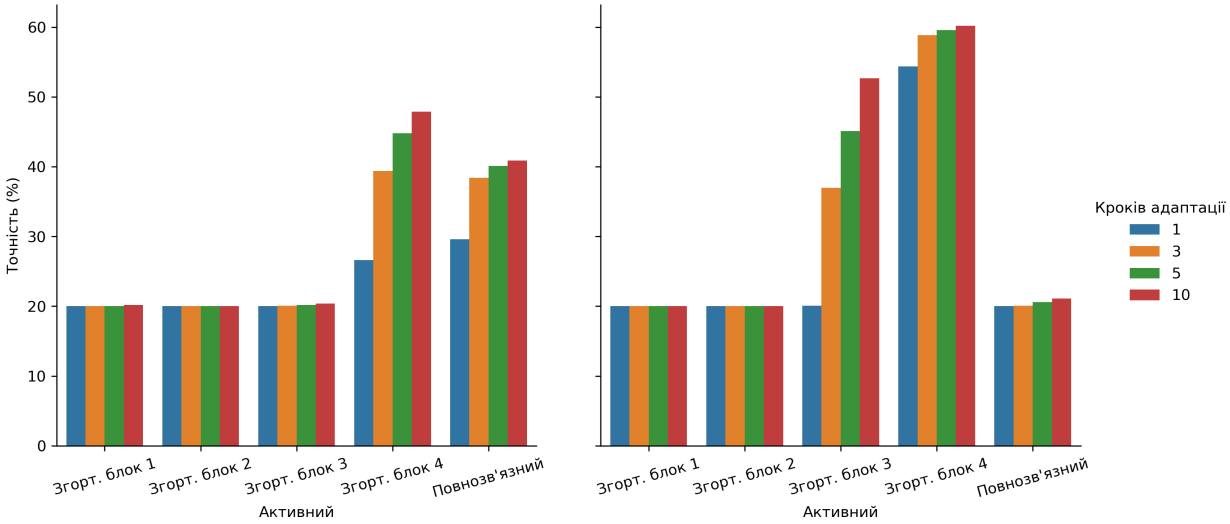
Для експериментів використовувався новий набір даних CIFAR-FS [188] для навчання за декількома прикладами. Набір даних створено на основі відомого класифікаційного набору даних CIFAR-100 [154], який містить зображення різних видів ссавців, рептилій, квітів, створених людиною речей тощо. Зображення кольорові та мають розмір 32×32.

При звичайному навчанні нейронних мереж всі 100 класів набору даних CIFAR-100 мають бути рівномірно представлені в навчальній, перевірній та тестовій вибірках. Натомість при навчанні за кількома прикладами класи розбиті між наборами для навчання, перевірки та тестування, куди включено 64, 16 і 20 класів відповідно. Точні класи, які входять до кожного розбиття, важливі, і для CIFAR-FS визначені в [188]. Використовуючи різні класи для навчання та тестування, можна краще оцінити якість адаптації до нових класів. Набір даних CIFAR-FS було взято для експериментів, оскільки він не аналізувався авторами MAML. Точність роботи методу, яку наведено далі обчислено на тестовому наборі.

Усі конфігурації навчання та заміри часу було проведено на розробленій в рамках даної роботи реалізації методу MAML і протестовано на графічному процесорі NVIDIA GTX 1050Ti.

5.2.2 Аналіз якості та часу виконання розробленого методу на тривіальних Λ-шаблонах

У цьому розділі досліджено поведінку CNN4 для тривіальних шаблонів Λ, таких, що $(\exists! l : \Lambda_l = 1)$, тобто під час процедури адаптації оновлюється лише один шар мережі. На рис. 5.2а та 5.2б показано результати експерименту для 1-прикл. 5-клас. та 5-прикл. 5-клас. конфігурацій відповідно. Аби ви-



(а) 1-прикл. 5-клас. (б) 5-прикл. 5-клас.

Рис. 5.2. Точність адаптації для тривіальних шаблонів Λ (5-клас.)

явити вплив кількості кроків адаптації, показано точність для $P = 1, 3, 5, 10$ кроків адаптації. Як видно, точність моделі суттєво відрізняється залежно від активного шару: в налаштуванні 1-прикл. 5-клас. корисно навчати згортковий блок 4 та повнозв'язний шари, в режимі 5-прикл. 5-клас – згорткові блоки 3 та 4. Навчання всіх інших шарів не має позитивного впливу на якість, точність залишається на рівні випадкового вгадування (20 %).

У випадку 1-прикл. 5-клас. навчання лише одного з перших трьох згорткових шарів не має ефекту, точність залишається на рівні випадкового вгадування (20 %). Однак навчання або згорткового шару №4, або останнього повнозв'язного покращує точність моделі. Зауважимо, що кількість параме-

трів у шарах відрізняється (табл. 5.1), а кінцевий шар має різну кількість параметрів залежно від кількості N вихідних класів. Перший згортковий блок

Таблиця 5.1. Кількість параметрів у кожному шарі CNN4

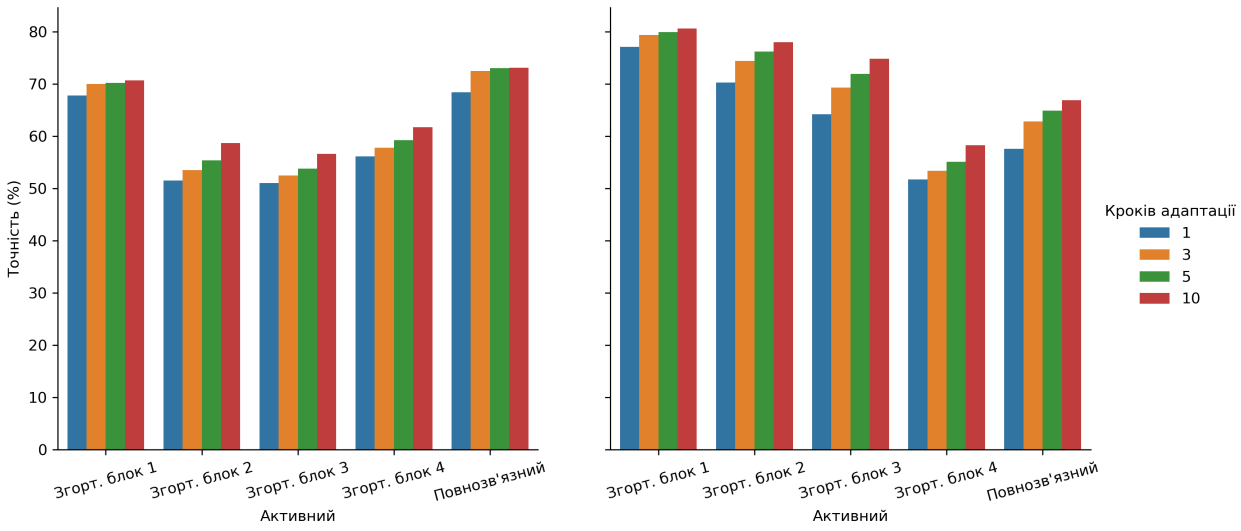
Назва шару	К-сть параметрів
Згорт. блок 1	960
Згорт. блок 2	9 312
Згорт. блок 3	9 312
Згорт. блок 4	9 312
Повнозв'язний	
2-клас.	1 602
5-клас.	4 005
Загалом	
2-клас.	30 498
5-клас.	32 901

і кінцевий повнозв'язний шари мають меншу кількість параметрів, ніж внутрішні згорткові блоки.

У сценарії 5-прикл. 5-клас. лише навчання згорткових шарів 3 і 4 має позитивний вплив на якість, якщо адаптувати лише один шар окремо. Легко бачити, що кількість кроків адаптації має значний вплив на якість, коли ввімкнено лише згортковий шар №3. Вплив є більш суттєвим, ніж коли під час адаптації оновлюються всі шари мережі.

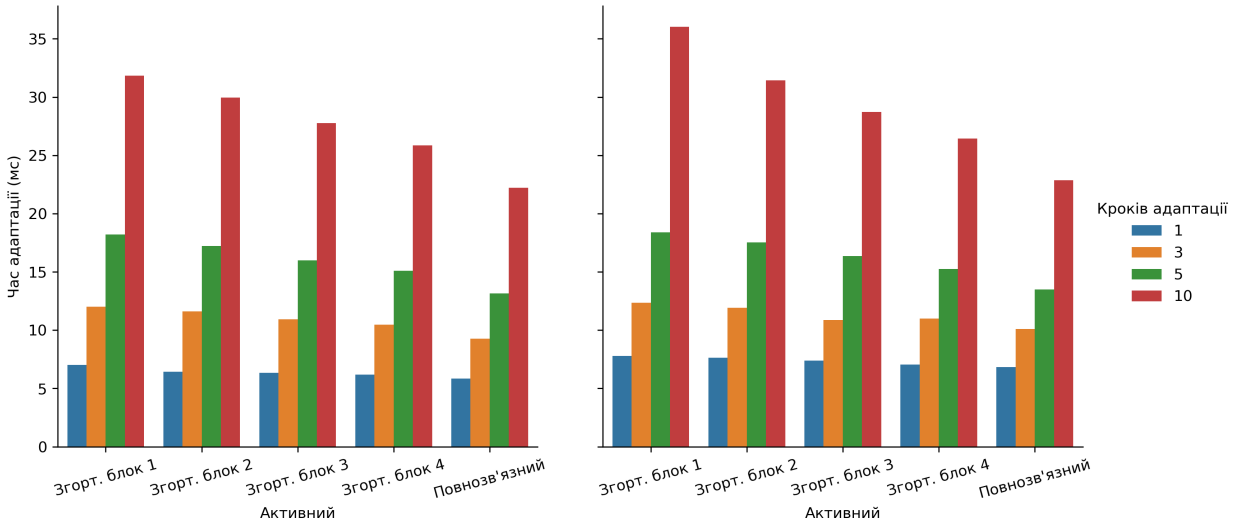
На рис. 5.3а та 5.3б подібний експеримент зображено для 1-прикл. 2-клас. та 5-прикл. 2-клас. конфігурацій відповідно. Рівень випадкового вгадування для цих конфігурацій тепер становить 50 %. На відміну від попередніх експериментів, оновлення перших шарів позитивно впливає на результуючу точність, тоді як оновлення виключно згорткового блоку 4 не забезпечує найкращих результатів у будь-якій конфігурації.

На рис. 5.4 показано час адаптації для 1-прикл. 5-клас. та 5-прикл. 5-клас. конфігурацій для всіх тривіальних Λ-шаблонів. Як видно, обраний шаблон



(а) 1-прикл. 2-клас. (б) 5-прикл. 2-клас.

Рис. 5.3. Точність адаптації для тривіальних Λ-шаблонів (2-клас.)



(а) 1-прикл. 5-клас. (б) 5-прикл. 5-клас.

Рис. 5.4. Час адаптації для тривіальних Λ-шаблонів (5-клас.)

Λ має значний вплив на швидкість адаптації. Для того, щоб оновити ваги в деякому шарі, потрібно обчислити градієнти для всіх шарів після нього. Але обчислення градієнта можна пропустити для деякого шару, якщо ні сам шар, ні попередні шари не потребують оновлення параметрів, а отже, оновлення перших шарів відбувається повільніше, ніж оновлення останніх. Аналогічна поведінка спостерігається в 2-клас. конфігураціях.

5.2.3 Пошук оптимального Λ-шаблону за заданої цільової якості роботи методу

У цьому підрозділі виконано пошук оптимального Λ-шаблону за заданої цільової якості роботи методу.

В таблиці 5.2 наведено точність роботи та середній час адаптації мережі CNN4, навченої методом MAML із певною кількістю кроків адаптації P при падінні точності не більше 3 %. В разі необхідності, пошук Λ-шаблонів можна виконати і при більшому пороговому значенню падіння точності. Дані впорядковані від найшвидшого до найповільнішого. Аналіз цих результатів дав змогу виявити шаблон $\Lambda = \{1, 0, 1, 1, 1\}$, за якого процедура адаптації прискорюється на 7,51 % із втратою точності у 0,33 % (усереднено за чотирма сценаріями тестування). При використанні шаблона $\Lambda = \{0, 1, 1, 1, 1\}$ можна досягти прискорення на 14,96 % при втратах якості у 1,25 %. Гірша якість останнього шаблону відповідає попередньому обговоренню рис. 5.3, де було показано, що ввімкнення першого шару згорткової мережі має важливе значення для точності 2-клас. навчання. Ефективність шаблонів (відношення середньої точності до часу) дорівнює 107,03 % та 112,95 % відповідно, якщо за 100 % брати повний Λ-шаблон.

Також на основі проведених експериментів можна відзначити, що на наборі даних CIFAR-FS зменшення кількості кроків адаптації з 10 до 5 при-

Таблиця 5.2. Прискорення адаптації в залежності від шаблону Λ і кількості кроків адаптації

Кроків адапт., P	Шаблон, Λ	Точність (%)				Середн. час адаптації (мс)	Приск. (%)
		2-клас.		5-клас.			
		1-прикл.	5-прикл.	1-прикл.	5-прикл.		
3	0,1,1,1,1	74,7	83,2	49,3	69,7	13,3	212,03
3	1,0,1,1,1	76,6	85,9	49,3	69,8	13,9	198,56
3	1,1,1,1,1	76,6	87,2	49,3	70,0	15,0	176,67
5	0,1,1,1,1	75,2	83,9	51,5	69,9	20,0	107,50
5	1,0,1,1,1	76,9	86,2	51,4	70,1	21,1	96,68
5	1,1,1,1,1	77,0	87,4	51,6	70,2	22,6	83,63
10	0,1,1,1,1	75,4	84,6	51,7	70,1	36,1	14,96
10	1,0,1,1,1	77,1	86,6	51,7	70,1	38,6	7,51
10	1,1,1,1,1	77,2	87,6	51,7	70,3	41,5	0,00

звело до падіння якості класифікації на 0,15 %, тому рекомендованим є виконання 5 або 3 кроків адаптації. При такій кількості кроків адаптації зазначені Λ-шаблони також можна використовувати. При $P = 3$ прискорення сягає 198,56 % ($\Lambda = \{1, 0, 1, 1, 1\}$, падіння точності: 1,30 %) та 212,03 % ($\Lambda = \{0, 1, 1, 1, 1\}$, падіння точності: 2,48 %)

5.2.4 Покращення точності із Λ-шаблонами за один крок адаптації

За результатами проведених експериментів виявилось, що у випадку навчання з одним кроком адаптації ($P = 1$) оновлення лише частини параметрів нейронної мережі підвищує точність методу MAML. Результати показано у табл. 5.3. Найкращий результат виділено жирним. Найбільш значне покращення якості досягнуто для 5-прикл. 5-клас. конфігурації із шаблоном $\Lambda = \{1, 1, 0, 1, 0\}$, де точність звичайного MAML залишалась близькою до рівня випадкового вгадування, а мережа із Λ-шаблоном $\{1, 1, 0, 1, 0\}$ досягла якості у 54,8 %.

Однією з причин, чому оновлення параметрів одних шарів нейронної ме-

Таблиця 5.3. Покращення точності класифікації за $P = 1$ крок адаптації із вибором Λ-шаблону

	1-прикл. 2-клас.	5-прикл. 2-клас.	1-прикл. 5-клас.	5-прикл. 5-клас.
Точність $\Lambda = \{1, 1, 1, 1, 1\}$	74,3 %	86,0 %	36,8 %	20,4 %
Точність на обраному Λ	**74,5 %**	**86,2 %**	**36,9 %**	**54,8 %**
Обраний Λ-шаблон	1,1,1,0,1	1,1,1,0,1	1,1,0,1,1	1,1,0,1,0

режі і «заморожування» інших покращує якість для сценарію однокрокової адаптації, може бути те, що різні шари нейронних мереж вивчають ознаки, які відрізняються за складністю: чим ближче шар до входу, тим простіші ознаки, що було показано в [189]. Перший шар вчиться виявляти, наприклад, такі елементи, як краї, лінії або градієнти кольорів. Другий шар виявляє складніші форми, приміром, коло або смуги, тоді як останні шари вивчають ознаки високого рівня, такі як очі, обличчя, об'єкти, схожі на текст і т.д. Які саме ознаки було вивчено, очевидно, залежить від навчального набору даних. Деякі з цих ознак мало змінюються між задачами, а отже їх має сенс залишати незмінними при адаптації до нових задач.

Крім того, автори Meta-SGD [150] показали, що, вивчаючи окремі коефіцієнти швидкості навчання для кожного з параметрів мережі, можна перевершити якість методу MAML. Однак Meta-SGD уповільнив навчання, оскільки необхідно було вивчати як ваги, так і швидкості навчання. Навпаки, час навчання в запропонованому підході із Λ-шаблонами не змінюється. Заморожування деяких шарів нейронної мережі дозволяє зменшити кількість обчислень, необхідних на етапі адаптації.

Використовуючи описані характеристики ознак, які вивчають різні шари мережі, далі наведений аналіз відмінностей в якості після оновлення різних шарів на рисунках 5.2 та 5.3. При класифікації за кількома прикладами задачі відрізняються типами об'єктів, які має класифікувати мережа, а навчальні та

тестові набори включають класи, що не перетинаються. Відтак, можна було б очікувати, що лише останні шари мережі мусять змінюватися для адаптації до нових задач і класів – це те, що можна побачити у випадку 5-класової класифікації, як показано на рис. 5.2. Однак, таке твердження суперечить результатам експерименту з рис. 5.3.

Щоб зрозуміти причину протиріччя, слід помітити, що класи зображень в наборі даних CIFAR-FS відносяться до декількох надкласів: наприклад, «бобер», «дельфін», «видра», «тюлень», «кит» належать до надкласу «водні ссавці». Інші приклади надкласів включають «риби», «великі хижаки», «побутові електроприлади» тощо. Отже, загалом існує невелика кількість груп із подібними класами всередині їх. З цих прикладів стає очевидним, що зображення з різних надкласів мають істотно різну колірну гамму. Фон зображень водних ссавців і риб зазвичай містить синій і сірий кольори, тоді як у великих хижих тварин в ньому може бути більше жовтого і зеленого. У разі 2-класової класифікації більш імовірно, ніж в 5-класовій, що мережа буде вирішувати задачу класифікувати між підкласами одного надкласу. Отже, якщо перший шар оновлено в 2-клас. сценарії навчання з кількома прикладами, нові ваги краще пристосовуються до зображень з іншою колірною гамою.

Висновки до розділу 5

В даному розділі запропоновано метод Λ-шаблонів прискорення навчання нейронної мережі для класифікації зображень за кількома прикладами. Його реалізація дозволила зменшити кількість обчислень, необхідних для адаптації ваг мережі до нових класів. Підбираючи шаблон адаптації до нових класів її час було зменшено на 7,51 % (падіння точності: 0,33 %), або на 14,96 % (падіння точності: 1,25 %) за умови незмінної кількості кроків адаптації (10 кроків). При використання 3 кроків адаптації у поєднанні із Λ шаблонами

досягнено прискорення на 198,56 % (падіння точності: 1,30 %) та 212,03 % (падіння точності: 2,48 %).

Також Λ-шаблони покращили якість методу у випадку однокрокової адаптації ваг мережі на всіх конфігураціях. Найбільше покращення отримано в конфігурації по 5 прикладів на 5 класів, де оригінальний метод MAML за один крок адаптації демонстрував точність у 20,4 %, що є показником близьким до випадкового вгадування. В той самий час, вибір Λ-шаблону дозволив вирішити цю проблему, збільшуючи результуючу якість методу до 54,8 %.

Зменшення часу роботи у порівнянні з оригінальним методом MAML дозволить будувати більш ефективні додатки, де часто необхідно проводити адаптацію нейронної мережі для розпізнавання нових класів. Можливим впровадженням запропонованого методу Λ-шаблонів прискорення навчання може слугувати застосунок, котрий дозволить користувачам швидко налаштувати мережу для задач класифікації зображень через веб-інтерфейс лише за кількома прикладами на клас.

ВИСНОВКИ

У дисертаційній роботі розв’язано задачу прискорення навчання і виконання згорткових нейронних мереж для задач класифікації та сегментації зображень без втрат (або з якомога меншими втратами) якості розпізнавання за рахунок розробки змінюваних нейронних мереж і методів їх навчання. У межах проведеного дослідження отримано такі основні результати:

1. Розглянуто архітектури згорткових нейронних мереж для задач класифікації та сегментації, проаналізовано їх особливості, зокрема обчислювальну складність та кількість параметрів, виявлено переваги та недоліки. Зазначено, що з точки зору проблем базових архітектур нейронних мереж важливим є зменшення часу виконання за мінімальних втрат якості та можливість зміни архітектури нейронної мережі для обчислення на пристроях із різними обчислювальними можливостями. Обґрунтовано вибір архітектур для основи дисертаційного дослідження.

2. Для задачі класифікації розроблено змінювану згорткову нейронну мережу (мережу PTA) та метод її навчання, які дозволяють змінювати архітектуру мережі під час або після її навчання з урахуванням обчислювальних можливостей пристроїв, на яких вона розгортається. За результатами проведених експериментальних досліджень на наборі даних ImageNet мережа PTA зайняла п’яте місце серед 17 провідних архітектур мереж за співвідношенням якість розпізнавання/час виконання. Зокрема час виконання мережі у порівнянні з оригінальною MobileNetV2 зменшено на 13,74 % при падінні точності (топ 1) на 3,68 %. На наборі даних CelebA-Spoof мережа PTA перевершує оригінальну за всіма метриками та дозволяє зменшити час виконання до 20 %. Найкращі результати за метриками точності і BPCER має конфігурація PTA-LLL із значеннями (в дужках – результати MobileNetV2) у 97,85 % (проти

96,74 %) і 1,98 % (проти 4,18 %) відповідно; за APCER – PTA-LHH з 0,70 % (проти 1,07 %); за ACER – PTA-BBB з 2,13 % (проти 2,63 %). Загальний час навчання PTA моделі зменшено на 14,34 % порівняно із MobileNetV2.

3. Змінювану згорткову мережу застосовано в підсистемі антиспуфінгу мобільної системи контролю доступу із RFID мітками, де аналіз зображень виконується безпосередньо на смартфоні користувача із метою зменшення навантаження на сервер та підвищення захищеності самої системи контролю доступу. Система контролю доступу включає: адміністративну панель та систему моніторингу; мобільний додаток, що здійснює пошук облич та перевірку зображення на спуфінг; серверну програму, яка оброблює, зберігає та надає дані для додатків на ПК і смартфоні.

4. Розроблено архітектуру згорткової нейронної мережі для задачі сегментації за рахунок вбудови в мережу U-Net додаткового згорткового блоку PTA. Отриману змінювану згорткову мережу навчено на наборі даних CamVid. Мережа U-Net+PTA продемонструвала зменшення часу виконання мережі на 6,09 % у порівнянні з оригінальною мережею U-Net. Мережу U-Net+PTA розгорнуто на крайовому, мобільних, персональних комп'ютерах та графічному процесорі. Показано, що остаточну навчену мережу PTA можна перемикати під час виконання між шістьма конфігураціями, які мають вищу якість, ніж оригінальна мережа U-Net ($Dice_{score}$ = 0,8583). Найкращий результат за швидкістю виконання продемонструвала PTA-LLL (прискорення 6,09 %, $Dice_{score}$ = 0,8647).

5. Розроблено метод Λ-шаблонів прискорення оптимізаційного мета-навчання, який на відміну від існуючих дозволяє змінювати кількість обчислень у методі зворотного розповсюдження помилки, за рахунок чого пришвидшено адаптацію нейронної мережі до нових класів за малою кількістю прикладів на 7,51 % при втратах якості у 0,33 %. Удосконалено метод MAML за рахунок впровадження Λ-шаблонів, що дозволило

підвищити ефективність процедури адаптації до нових класів за малою кількістю навчальних прикладів від 7,03 % до 12,95 % в залежності від конфігурації.

Результати, отримані у дисертаційній роботі, можуть бути використані під час розв’язання задач класифікації та сегментації зображень для прискорення навчання згорткових нейронних мереж, а також їх виконання на пристроях із різними обчислювальними можливостями без або за мінімальних втрат якості розпізнавання; для навчання мереж за малою кількістю навчальних прикладів; в системах відстеження рухів та анімації обличчя тощо; в системах контролю і управління доступом до технологічного обладнання, дверей тощо.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. *Khabarlak K.* Post-Train Adaptive MobileNet for Fast Anti-Spoofing // Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security, Khmelnytskyi, Ukraine, March 23–25. T. 3156. — CEUR-WS.org, 2022. — С. 44–53. — (CEUR Workshop Proceedings). — URL: http://ceur-ws.org/Vol-3156/keynote5.pdf.

2. *Хабарлак К. С.* Адаптація мета-навчання на частковому шаблоні // Матеріали VII міжнародної науково-технічної конференції Комп'ютерне моделювання та оптимізація складних систем. — Дніпро, 11.2021. — С. 121–122.

3. *Хабарлак К. С.* Аналіз шаблонів адаптації мета-навчання // Матеріали IX Всеукраїнської науково-технічної конференції студентів, аспірантів та молодих вчених «Молодь: наука та інновації». — Дніпро, 11.2021. — С. 328–329.

4. *Хабарлак К. С.* Прискорене навчання нейронної мережі за декількома прикладами // XVI Міжнародна конференція з проблем використання інформаційних технологій в освіті, науці та промисловості. — Дніпро, 12.2021. — С. 62–64.

5. *Хабарлак К. С.* Зміна архітектури нейронної мережі після навчання // Тези дев'ятої міжнародної науково-технічної конференції Інформатика, управління та штучний інтелект. — Харків – Краматорськ, 05.2022. — С. 135.

6. *Хабарлак К. С.* Нейромережний пошук об'єктів на мобільних пристроях // Матеріали XII Всеукраїнської науково-технічної конференції

студентів, аспірантів та молодих вчених «Наукова весна». — Дніпро, 05.2022. — С. 171.

7. *Хабарлак К. С.* Адаптивна після навчання нейронна мережа // Тези V Всеукраїнської Інтернет-конференції здобувачів вищої освіти і молодих учених Інформаційні технології: теорія і практика. — Запоріжжя, 06.2022. — С. 20–21.

8. *Хабарлак К. С.* Нейро-мережева система класифікації із конфігурацією після навчання // Матеріали X Міжнародної науково-технічної конференції студентів, аспірантів і молодих вчених «Молодь: наука та інновації». — Дніпро, 11.2022. — С. 383.

9. *Khabarlak K.* Semantic segmentation with Post-Train Adaptive Neural Network // Тези XI міжнародної науково-практичної конференції «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій». — Запоріжжя, 12.2022. — С. 124–125.

10. *Хабарлак К. С.* Конфігурація після навчання нейронної мережі для сегментації зображень // Матеріали XIII Міжнародної науково-технічної конференції аспірантів та молодих вчених «Наукова весна». — Дніпро, 03.2023. — С. 194–195.

11. *Хабарлак К. С.* Проблеми нейронних мереж для розпізнавання на пристроях із різними обчислювальними можливостями // Тези VI Всеукраїнської науково-практичної Інтернет-конференції здобувачів вищої освіти і молодих учених Інформаційні технології: теорія і практика. — Харків, 03.2023. — С. 101–102.

12. *Khabarlak K. S.* Faster Optimization-Based Meta-Learning Adaptation Phase // Radio Electronics, Computer Science, Control. — 2022. —

Квіт. — № 1. — С. 82–92. — DOI: 10.15588/1607-3274-2022-1-10. — URL: http://ric.zntu.edu.ua/article/view/254615.

13. *Khabarlak K.* Post-Train Adaptive U-Net for Image Segmentation // Information Technology: Computer Science, Software Engineering and Cyber Security. — 2022. — № 2. — С. 73–78. — DOI: 10.32782/IT/2022-2-8. — URL: https://doi.org/10.32782/IT/2022-2-8.

14. *Хабарлак К. С.* Особливості роботи методів пошуку облич на мобільних пристроях // System Technologies. — 2021. — Т. 6, № 137. — С. 34–45. — DOI: 10.34185/1562-9945-6-137-2021-04. — URL: https://journals.nmetau.edu.ua/index.php/st/issue/view/118/97.

15. Комп'ютерна програма «Мобільна нейромережева система пошуку облич із анті-спуфінгом» : авт. свід. України №110917 / К. С. Хабарлак. — 11.01.2022.

16. *Khabarlak K.* Mobile Application for RFID Access Control System // V міжнародна науково-практична конференція «Прикладні науково-технічні дослідження». — Івано-Франківськ, 04.2021. — С. 99–100.

17. *Хабарлак К. С.* Анти-спуфінг для системи контролю доступу із RFID мітками // Збірник матеріалів III Всеукраїнської конференції «Теоретико-практичні проблеми використання математичних методів і комп'ютерно-орієнтованих технологій в освіті та науці». — Київ, 04.2021. — С. 141–142.

18. *Хабарлак К. С.* On Face Detection and Anti-Spoofing in Mobile Access Control // Тези доповідей VIII Міжнародної науково-практичної конфе-

ренції «Інформаційні технології в освіті, науці і виробництві (ІТОНВ-2021)». — Луцьк, 05.2021. — С. 181–183.

19. *Хабарлак К. С.* Про адаптацію мета-навчання нейронних мереж // Матеріали міжнародної конференції II International Scientific Symposium “Intelligent Solutions-S”. — Ужгород, 09.2021. — С. 81.

20. *Khabarlak K.*, *Koriashkina L.* Fast Facial Landmark Detection and Applications: A Survey // Journal of Computer Science and Technology. — 2022. — Квіт. — Т. 22, № 1. — С. 12–41. — DOI: 10.24215/16666038.22.e02. — URL: https://journal.info.unlp.edu.ar/JCST/article/view/1972.

21. *Khabarlak K. S.*, *Koriashkina L. S.* Scoping Adversarial Attack for Improving Its Quality // Radio Electronics, Computer Science, Control. — 2019. — Трав. — № 2. — С. 108–118. — DOI: 10.15588/1607-3274-2019-2-12. — URL: http://ric.zntu.edu.ua/article/view/178284.

22. *Khabarlak K. S.*, *Koriashkina L. S.* Mobile Access Control System Based on RFID Tags and Facial Information // Bulletin of National Technical University ”KhPI”. Series: System Analysis, Control and Information Technologies. — 2020. — Т. 2, № 4. — С. 69–74. — DOI: 10.20998/2079-0023.2020.02.12. — URL: http://samit.khpi.edu.ua/article/view/2079-0023.2020.02.12.

23. *Хабарлак К. С.*, *Коряшкіна Л. С.* Деякі особливості гіперпараметрів глибоких нейронних мереж // III Всеукраїнська Інтернет-конференція здобувачів вищої освіти і молодих учених «Інформаційні технології: теорія і практика». — Харків, 03.2020. — С. 98–99.

24. *Khabarlak K.*, *Koriashkina L.* Top image classification accuracy through hyperparameter search // 15th International Forum for Students and Young Researchers “Widening our horizons”. — Dnipro, 05.2020. — С. 271–274.

25. *Хабарлак К. С.*, *Коряшкіна Л. С.* Тестування швидкості виконання алгоритмів пошуку обличчь для системи контролю доступу // VII Всеукраїнська науково-технічна конференція молодих учених, аспірантів та студентів «Автоматизація, контроль та управління: пошук ідей та рішень» (АКУ-2021). — Покровськ, 05.2021. — С. 60–61.

26. *Brostow G. J.*, *Fauqueur J.*, *Cipolla R.* Semantic object classes in video: A high-definition ground truth database // Pattern Recognit. Lett. — 2009. — Т. 30, № 2. — С. 88–97. — DOI: 10.1016/j.patrec.2008.04.005. — URL: https://doi.org/10.1016/j.patrec.2008.04.005.

27. Intelligent System for Building Separation on a Semantically Segmented Map / V. V. Hnatushenko [та ін.] // Proceedings of the 2nd International Workshop on Intelligent Information Technologies & Systems of Information Security with CEUR-WS, Khmelnytskyi, Ukraine, March 24-26, 2021. Т. 2853. — CEUR-WS.org, 2021. — С. 1–11. — (CEUR Workshop Proceedings). — URL: http://ceur-ws.org/Vol-2853/keynote1.pdf.

28. *Ronneberger O.*, *Fischer P.*, *Brox T.* U-Net: Convolutional Networks for Biomedical Image Segmentation // Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III. Т. 9351. — Springer, 2015. — С. 234–241. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-319-24574-4_28. — URL: https://doi.org/10.1007/978-3-319-24574-4%5C_28.

29. *Viola P. A.*, *Jones M. J.* Rapid Object Detection using a Boosted Cascade of Simple Features // 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), 8-14 December 2001, Kauai, HI, USA. — IEEE Computer Society, 2001. — С. 511–518. — DOI: 10.1109/CVPR.2001.990517.

30. *Dalal N.*, *Triggs B.* Histograms of Oriented Gradients for Human Detection // 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA. — IEEE Computer Society, 2005. — С. 886–893. — DOI: 10.1109/CVPR.2005.177.

31. ImageNet: A large-scale hierarchical image database / J. Deng [та ін.] // 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA. — IEEE Computer Society, 2009. — С. 248–255. — DOI: 10.1109/CVPR.2009.5206848.

32. *Krizhevsky A.*, *Sutskever I.*, *Hinton G. E.* ImageNet Classification with Deep Convolutional Neural Networks // Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States. — 2012. — С. 1106–1114. — URL: https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

33. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices / X. Zhang [та ін.] // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. — 2018. — С. 6848–6856. — DOI: 10.1109/CVPR.2018.00716.

34. MobileNetV2: Inverted Residuals and Linear Bottlenecks / M. Sandler [та ін.] // 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. — Computer Vision Foundation / IEEE Computer Society, 2018. — С. 4510–4520. — DOI: 10.1109/CVPR.2018.00474.

35. Searching for MobileNetV3 / A. Howard [та ін.] // 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. — IEEE, 2019. — С. 1314–1324. — DOI: 10.1109/ICCV.2019.00140.

36. MnasNet: Platform-Aware Neural Architecture Search for Mobile / M. Tan [та ін.] // IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. — Computer Vision Foundation / IEEE, 2019. — С. 2820–2828. — DOI: 10.1109/CVPR.2019.00293.

37. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size / F. N. Iandola [та ін.] // CoRR. — 2016. — Т. abs/1602.07360. — arXiv: 1602.07360. — URL: http://arxiv.org/abs/1602.07360.

38. BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs / V. Bazarevsky [та ін.] // CoRR. — 2019. — Т. abs/1907.05047. — arXiv: 1907.05047. — URL: http://arxiv.org/abs/1907.05047.

39. *Simonyan K.*, *Zisserman A.* Very Deep Convolutional Networks for Large-Scale Image Recognition // 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. — 2015. — URL: http://arxiv.org/abs/1409.1556.

40. Deep Residual Learning for Image Recognition / K. He [та ін.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. — IEEE Computer Society, 2016. — С. 770–778. — DOI: 10.1109/CVPR.2016.90.

41. *Ioffe S.*, *Szegedy C.* Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. Т. 37. — JMLR.org, 2015. — С. 448–456. — (JMLR Workshop and Conference Proceedings). — URL: http://proceedings.mlr.press/v37/ioffe15.html.

42. *Hu J.*, *Shen L.*, *Sun G.* Squeeze-and-Excitation Networks // 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. — Computer Vision Foundation / IEEE Computer Society, 2018. — С. 7132–7141. — DOI: 10.1109/CVPR.2018.00745.

43. Dynamic Neural Networks: A Survey / Y. Han [та ін.] // IEEE Trans. Pattern Anal. Mach. Intell. — 2022. — Т. 44, № 11. — С. 7436–7456. — DOI: 10.1109/TPAMI.2021.3117837. — URL: https://doi.org/10.1109/TPAMI.2021.3117837.

44. *Graves A.* Adaptive Computation Time for Recurrent Neural Networks // CoRR. — 2016. — Т. abs/1603.08983. — arXiv: 1603.08983. — URL: http://arxiv.org/abs/1603.08983.

45. Spatially Adaptive Computation Time for Residual Networks / M. Figurnov [та ін.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. — IEEE Computer Society, 2017. —

C. 1790–1799. — DOI: 10 . 1109 / CVPR . 2017 . 194. — URL: https://doi.org/10.1109/CVPR.2017.194.

46. CondConv: Conditionally Parameterized Convolutions for Efficient Inference / B. Yang [та ін.] // Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. — 2019. — С. 1305–1316. — URL: https://proceedings.neurips.cc/paper/2019/hash/f2201f5191c4e92cc5af043eebfd0946-Abstract.html.

47. *Ha D.*, *Dai A. M.*, *Le Q. V.* HyperNetworks // 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. — OpenReview.net, 2017. — URL: https://openreview.net/forum?id=rkpACe1lx.

48. Mixed Precision Training / P. Micikevicius [та ін.] // 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. — OpenReview.net, 2018. — URL: https://openreview.net/forum?id=r1gs9JgRZ.

49. Training Deep Neural Networks with 8-bit Floating Point Numbers / N. Wang [та ін.] // Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada. — 2018. — С. 7686–7695. — URL: https://proceedings.neurips.cc/paper/2018/hash/335d3d1cd7ef05ec77714a215134914c-Abstract.html.

50. *Bulat A.*, *Tzimiropoulos G.* Binarized Convolutional Landmark Localizers for Human Pose Estimation and Face Alignment with Limited Resources // IEEE International Conference on Computer Vision, ICCV 2017,

Venice, Italy, October 22-29, 2017. — IEEE Computer Society, 2017. — С. 3726–3734. — DOI: 10.1109/ICCV.2017.400.

51. Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference / B. Jacob [та ін.] // 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. — Computer Vision Foundation / IEEE Computer Society, 2018. — С. 2704–2713. — DOI: 10.1109/CVPR.2018.00286.

52. CondenseNet: An Efficient DenseNet Using Learned Group Convolutions / G. Huang [та ін.] // 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. — Computer Vision Foundation / IEEE Computer Society, 2018. — С. 2752–2761. — DOI: 10.1109/CVPR.2018.00291. — URL: http://openaccess.thecvf.com/content%5C_cvpr%5C_2018/html/Huang%5C_CondenseNet%5C_An%5C_Efficient%5C_CVPR%5C_2018%5C_paper.html.

53. Learning Efficient Convolutional Networks through Network Slimming / Z. Liu [та ін.] // IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. — IEEE Computer Society, 2017. — С. 2755–2763. — DOI: 10.1109/ICCV.2017.298. — URL: https://doi.org/10.1109/ICCV.2017.298.

54. A Dataset and Benchmark for Large-Scale Multi-Modal Face Anti-Spoofing / S. Zhang [та ін.] // IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. — Computer Vision Foundation / IEEE, 2019. — С. 919–928. — DOI: 10.1109/CVPR.2019.00101.

55. Searching Central Difference Convolutional Networks for Face Anti-Spoofing / Z. Yu [та ін.] // 2020 IEEE/CVF Conference on Computer

Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. — Computer Vision Foundation / IEEE, 2020. — С. 5294–5304. — DOI: 10.1109/CVPR42600.2020.00534.

56. CelebA-Spoof: Large-Scale Face Anti-spoofing Dataset with Rich Annotations / Y. Zhang [та ін.] // Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XII. Т. 12357. — Springer, 2020. — С. 70–85. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-030-58610-2_5. — URL: https://doi.org/10.1007/978-3-030-58610-2%5C_5.

57. Feature Pyramid Networks for Object Detection / T. Lin [та ін.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. — IEEE Computer Society, 2017. — С. 936–944. — DOI: 10.1109/CVPR.2017.106. — URL: https://doi.org/10.1109/CVPR.2017.106.

58. Deep High-Resolution Representation Learning for Human Pose Estimation / K. Sun [та ін.] // IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. — Computer Vision Foundation / IEEE, 2019. — С. 5693–5703. — DOI: 10.1109/CVPR.2019.00584.

59. *Newell A.*, *Yang K.*, *Deng J.* Stacked Hourglass Networks for Human Pose Estimation // Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII. Т. 9912. — Springer, 2016. — С. 483–499. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-319-46484-8_29.

60. Towards Efficient U-Nets: A Coupled and Quantized Approach / Z. Tang [та ін.] // IEEE Trans. Pattern Anal. Mach. Intell. — 2020. — Т. 42, № 8. — С. 2038–2050. — DOI: 10.1109/TPAMI.2019.2907634.

61. Deep High-Resolution Representation Learning for Visual Recognition / J. Wang [та ін.] // IEEE Trans. Pattern Anal. Mach. Intell. — 2021. — Т. 43, № 10. — С. 3349–3364. — DOI: 10.1109/TPAMI.2020.2983686.

62. Biometric Access Card: An Overview. — 2021. — URL: https://blog.mantratec.com/biometric-swipe-card-an-overview (дата зверн. 18.03.2023).

63. Access control system with RFID and biometric facial recognition : пат. США 11790385 / K. Kail, C. Williams, R. Kail. — 01.11.2007.

64. RFID Based Security and Access Control System / U. B. Farooq [та ін.] // International journal of engineering and technology. — 2014. — Т. 6, № 4. — С. 309–314. — DOI: 10.7763/IJET.2014.V6.718.

65. Mobile Access Control Guide. — 2023. — URL: https://www.getkisi.com/guides/mobile-access-control-guide (дата зверн. 18.03.2023).

66. Guide to Mobile Access Control. — 2023. — URL: https://www.swiftlane.com/access-control/guide-to-mobile-access-control/ (дата зверн. 18.03.2023).

67. NFC Compatibility. — 2020. — URL: https://www.shopnfc.com/en/content/7-nfc-compatibility (дата зверн. 12.10.2020).

68. NFC Tag Specs – Tag NFC. — 2020. — URL: https://www.tagnfc.com/en/info/11-nfc-tags-specs (дата зверн. 12.10.2020).

69. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks / K. Zhang [та ін.] // IEEE Signal Processing Letters. — 2016. — Т. 23, № 10. — С. 1499–1503. — DOI: 10.1109/LSP.2016.2603342.

70. *Li S.*, *Deng W.* Deep Facial Expression Recognition: A Survey // IEEE Transactions on Affective Computing. — 2020. — С. 1–1. — DOI: 10.1109/TAFFC.2020.2981446.

71. VDub: Modifying Face Video of Actors for Plausible Visual Alignment to a Dubbed Audio Track / P. Garrido [та ін.] // Comput. Graph. Forum. — Chichester, GBR, 2015. — Трав. — Т. 34, № 2. — С. 193–204. — ISSN 0167-7055. — DOI: 10.1111/cgf.12552.

72. Few-Shot Adversarial Learning of Realistic Neural Talking Head Models / E. Zakharov [та ін.] // 2019 IEEE/CVF International Conference on Computer Vision (ICCV). — Los Alamitos, CA, USA : IEEE Computer Society, 11.2019. — С. 9458–9467. — DOI: 10.1109/ICCV.2019.00955.

73. *Johnson J.*, *Alahi A.*, *Fei-Fei L.* Perceptual Losses for Real-Time Style Transfer and Super-Resolution // Computer Vision – ECCV 2016. — Cham : Springer International Publishing, 2016. — С. 694–711. — ISBN 978-3-319-46475-6.

74. A Neural Lip-Sync Framework for Synthesizing Photorealistic Virtual News Anchors / R. Zheng [та ін.] // 25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021. — IEEE, 2020. — С. 5286–5293. — DOI: 10.1109/ICPR48806.2021.9412187.

75. High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs / T. Wang [та ін.] // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — Los Alamitos, CA, USA : IEEE Computer Society, 06.2018. — С. 8798–8807. — DOI: 10.1109/CVPR.2018.00917.

76. FReeNet: Multi-Identity Face Reenactment / J. Zhang [та ін.] // 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2020. — С. 5325–5334. — DOI: 10.1109/CVPR42600.2020.00537.

77. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks / J.-Y. Zhu [та ін.] // 2017 IEEE International Conference on Computer Vision (ICCV). — 2017. — С. 2242–2251. — DOI: 10.1109/ICCV.2017.244.

78. Deepfakes and beyond: A Survey of face manipulation and fake detection / R. Tolosana [та ін.] // Information Fusion. — 2020. — Т. 64. — С. 131–148. — ISSN 1566-2535. — DOI: 10.1016/j.inffus.2020.06.014. — URL: https://www.sciencedirect.com/science/article/pii/S1566253520303110.

79. Driver Drowsiness Detection Model Using Convolutional Neural Networks Techniques for Android Application / R. Jabbar [та ін.] // 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT). — 2020. — С. 237–242. — DOI: 10.1109/ICIoT48696.2020.9089484.

80. Real-time monitoring of driver drowsiness on mobile platforms using 3D neural networks / J. S. Wijnands [та ін.] // Neural Computing and Applications. — 2019. — С. 1–13.

81. *Kim W.*, *Jung W.-S.*, *Choi H. K.* Lightweight Driver Monitoring System Based on Multi-Task Mobilenets // Sensors. — 2019. — Т. 19, № 14. — ISSN 1424-8220. — DOI: 10.3390/s19143200. — URL: https://www.mdpi.com/1424-8220/19/14/3200.

82. Driver Status Monitoring System in Autonomous Driving Era / T. Hyuga [та ін.] // OMRON TECHNICS. — 2019. — Т. 50.005EN. — URL: https://www.omron.com/global/en/assets/file/technology/omrontechnics/vol50/OMT_Vol50_005.pdf.

83. SSD: Single Shot MultiBox Detector / W. Liu [та ін.] // Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I. Т. 9905. — Springer, 2016. — С. 21–37. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-319-46448-0_2.

84. Going Deeper Into Face Detection: A Survey / S. Minaee [та ін.] // CoRR. — 2021. — Т. abs/2103.14983. — arXiv: 2103.14983. — URL: https://arxiv.org/abs/2103.14983.

85. Deep Face Recognition: A Survey / I. Masi [та ін.] // 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). — 2018. — С. 471–478. — DOI: 10.1109/SIBGRAPI.2018.00067.

86. *Schroff F.*, *Kalenichenko D.*, *Philbin J.* FaceNet: A unified embedding for face recognition and clustering // IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. — IEEE Computer Society, 2015. — С. 815–823. — DOI: 10.1109/CVPR.2015.7298682. — URL: https://doi.org/10.1109/CVPR.2015.7298682.

87. MobiFace: A Lightweight Deep Learning Face Recognition on Mobile Devices / C. N. Duong [та ін.] // 2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS). — 2019. — С. 1–6. — DOI: 10.1109/BTAS46853.2019.9185981.

88. MobileFaceNets: Efficient CNNs for Accurate Real-Time Face Verification on Mobile Devices / S. Chen [та ін.] // Biometric Recognition. — Cham : Springer International Publishing, 2018. — С. 428–438. — ISBN 978-3-319-97909-0.

89. *Ko B. C.* A Brief Review of Facial Emotion Recognition Based on Visual Information // Sensors. — 2018. — Т. 18, № 2. — ISSN 1424-8220. — DOI: 10.3390/s18020401. — URL: https://www.mdpi.com/1424-8220/18/2/401.

90. Facial feature point detection: A comprehensive survey / N. Wang [та ін.] // Neurocomputing. — 2018. — Т. 275. — С. 50–65. — ISSN 0925-2312. — DOI: 10.1016/j.neucom.2017.05.013. — URL: https://www.sciencedirect.com/science/article/pii/S0925231217308202.

91. *Wu Y.*, *Ji Q.* Facial Landmark Detection: A Literature Survey // Int. J. Comput. Vision. — USA, 2019. — Лют. — Т. 127, № 2. — С. 115–142. — ISSN 0920-5691. — DOI: 10.1007/s11263-018-1097-z.

92. LUVLi Face Alignment: Estimating Landmarks' Location, Uncertainty, and Visibility Likelihood / A. Kumar [та ін.] // 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2020. — С. 8233–8243. — DOI: 10.1109/CVPR42600.2020.00826.

93. *Wang X.*, *Bo L.*, *Fuxin L.* Adaptive Wing Loss for Robust Face Alignment via Heatmap Regression // 2019 IEEE/CVF International Conference on Computer Vision (ICCV). — 2019. — С. 6970–6980. — DOI: 10.1109/ICCV.2019.00707.

94. Look at Boundary: A Boundary-Aware Face Alignment Algorithm / W. Wu [та ін.] // 2018 IEEE/CVF Conference on Computer Vision and Pattern

Recognition. — 2018. — С. 2129–2138. — DOI: 10.1109/CVPR.2018.00227.

95. The Menpo benchmark for multi-pose 2D and 3D facial landmark localisation and tracking / J. Deng [та ін.] // International Journal of Computer Vision. — 2018. — С. 1–26.

96. Offline Deformable Face Tracking in Arbitrary Videos / G. G. Chrysos [та ін.] // 2015 IEEE International Conference on Computer Vision Workshop, ICCV Workshops 2015, Santiago, Chile, December 7-13, 2015. — IEEE Computer Society, 2015. — С. 954–962. — DOI: 10.1109/ICCVW.2015.126.

97. The First Facial Landmark Tracking in-the-Wild Challenge: Benchmark and Results / J. Shen [та ін.] // 2015 IEEE International Conference on Computer Vision Workshop, ICCV Workshops 2015, Santiago, Chile, December 7-13, 2015. — IEEE Computer Society, 2015. — С. 1003–1011. — DOI: 10.1109/ICCVW.2015.132.

98. *Tzimiropoulos G.* Project-Out Cascaded Regression with an application to face alignment // IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. — IEEE Computer Society, 2015. — С. 3659–3667. — DOI: 10.1109/CVPR.2015.7298989.

99. 300 Faces in-the-Wild Challenge: The First Facial Landmark Localization Challenge / C. Sagonas [та ін.] // 2013 IEEE International Conference on Computer Vision Workshops. — 2013. — С. 397–403. — DOI: 10.1109/ICCVW.2013.59.

100. Interactive Facial Feature Localization / V. Le [та ін.] // Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence,

Italy, October 7-13, 2012, Proceedings, Part III. T. 7574. — Springer, 2012. — C. 679–692. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-642-33712-3_49.

101. XM2VTSDB: The Extended M2VTS Database / K. Messer [та ін.] // : Second international conference on audio and video-based biometric person authentication. T. 964. — 1999. — C. 965–966.

102. Face Alignment Across Large Poses: A 3D Solution / X. Zhu [та ін.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. — IEEE Computer Society, 2016. — C. 146–155. — DOI: 10.1109/CVPR.2016.23.

103. Annotated Facial Landmarks in the Wild: A large-scale, real-world database for facial landmark localization / M. Köstinger [та ін.] // 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops). — 2011. — C. 2144–2151. — DOI: 10.1109/ICCVW.2011.6130513.

104. *Burgos-Artizzu X. P.*, *Perona P.*, *Dollár P.* Robust Face Landmark Estimation under Occlusion // 2013 IEEE International Conference on Computer Vision. — 2013. — C. 1513–1520. — DOI: 10.1109/ICCV.2013.191.

105. *Ghiasi G.*, *Fowlkes C. C.* Occlusion Coherence: Detecting and Localizing Occluded Faces // CoRR. — 2015. — T. abs/1506.08347. — arXiv: 1506.08347. — URL: http://arxiv.org/abs/1506.08347.

106. The Menpo facial landmark localisation challenge: A step towards the solution / S. Zafeiriou [та ін.] // Computer Vision and Pattern Recognition (CVPR) Workshops. — 2017.

107. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design / N. Ma [та ін.] // Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV. T. 11218. — Springer, 2018. — C. 122–138. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-030-01264-9_8.

108. Dense Face Alignment / Y. Liu [та ін.] // 2017 IEEE International Conference on Computer Vision Workshops (ICCVW). — 2017. — C. 1619–1628. — DOI: 10.1109/ICCVW.2017.190.

109. Style Aggregated Network for Facial Landmark Detection / X. Dong [та ін.] // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. — 2018. — C. 379–388. — DOI: 10.1109/CVPR.2018.00047.

110. Aggregation via Separation: Boosting Facial Landmark Detector With Semi-Supervised Style Translation / S. Qian [та ін.] // 2019 IEEE/CVF International Conference on Computer Vision (ICCV). — 2019. — C. 10152–10162. — DOI: 10.1109/ICCV.2019.01025.

111. Wing Loss for Robust Facial Landmark Localisation with Convolutional Neural Networks / Z. Feng [та ін.] // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — Los Alamitos, CA, USA : IEEE Computer Society, 06.2018. — C. 2235–2245. — DOI: 10.1109/CVPR.2018.00238.

112. PFLD: A Practical Facial Landmark Detector / X. Guo [та ін.] // CoRR. — 2019. — T. abs/1902.10859. — arXiv: 1902.10859. — URL: http://arxiv.org/abs/1902.10859.

113. *Bulat A.*, *Tzimiropoulos G.* How Far are We from Solving the 2D & 3D Face Alignment Problem? (and a Dataset of 230,000 3D Facial

Landmarks) // 2017 IEEE International Conference on Computer Vision (ICCV). — 2017. — С. 1021–1030. — DOI: 10.1109/ICCV.2017.116.

114. MobileFAN: Transferring deep hidden representation for face alignment / Y. Zhao [та ін.] // Pattern Recognition. — 2020. — Т. 100. — С. 107114. — ISSN 0031-3203. — DOI: 10.1016/j.patcog.2019.107114. — URL: https://www.sciencedirect.com/science/article/pii/S0031320319304157.

115. Robust Facial Landmark Detection via Aggregation on Geometrically Manipulated Faces / S. M. Iranmanesh [та ін.] // 2020 IEEE Winter Conference on Applications of Computer Vision (WACV). — 2020. — С. 319–329. — DOI: 10.1109/WACV45572.2020.9093508.

116. Structured Landmark Detection via Topology-Adapting Deep Graph Learning / W. Li [та ін.] // Computer Vision – ECCV 2020. — Cham : Springer International Publishing, 2020. — С. 266–283. — ISBN 978-3-030-58545-7.

117. PropagationNet: Propagate Points to Curve to Learn Structure Information / X. Huang [та ін.] // 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. — Computer Vision Foundation / IEEE, 2020. — С. 7263–7272. — DOI: 10.1109/CVPR42600.2020.00729.

118. Improving Robustness of Facial Landmark Detection by Defending Against Adversarial Attacks / C. Zhu [та ін.] // Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). — 10.2021. — С. 11751–11760.

119. LDDMM-Face: Large Deformation Diffeomorphic Metric Learning for Flexible and Consistent Face Alignment / H. Yang [та ін.] //

CoRR. — 2021. — T. abs/2108.00690. — arXiv: 2108.00690. — URL: https://arxiv.org/abs/2108.00690.

120. AnchorFace: An Anchor-based Facial Landmark Detector Across Large Poses / Z. Xu [та ін.] // Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021. — AAAI Press, 2021. — С. 3092–3100. — URL: https://ojs.aaai.org/index.php/AAAI/article/view/16418.

121. *Jin H.*, *Liao S.*, *Shao L.* Pixel-in-Pixel Net: Towards Efficient Facial Landmark Detection in the Wild // International Journal of Computer Vision. — 2021. — Груд. — Т. 129, № 12. — С. 3174–3194. — ISSN 1573-1405. — DOI: 10.1007/s11263-021-01521-4.

122. ADNet: Leveraging Error-Bias Towards Normal Direction in Face Alignment / Y. Huang [та ін.] // Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). — 10.2021. — С. 3080–3090.

123. *Lan X.*, *Hu Q.*, *Cheng J.* Revisting Quantization Error in Face Alignment // Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops. — 10.2021. — С. 1521–1530.

124. *Bulat A.*, *Sanchez E.*, *Tzimiropoulos G.* Subpixel Heatmap Regression for Facial Landmark Localization // CoRR. — 2021. — Т. abs/2111.02360. — arXiv: 2111.02360. — URL: https://arxiv.org/abs/2111.02360.

125. Generative Adversarial Networks / I. J. Goodfellow [та ін.] // CoRR. — 2014. — Т. abs/1406.2661. — arXiv: 1406 . 2661. — URL: http : // arxiv.org/abs/1406.2661.

126. Structured Feature Learning for Pose Estimation / X. Chu [та ін.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. — IEEE Computer Society, 2016. — С. 4715–4723. — DOI: 10.1109/CVPR.2016.510.

127. End-to-End Learning of Deformable Mixture of Parts and Deep Convolutional Neural Networks for Human Pose Estimation / W. Yang [та ін.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. — IEEE Computer Society, 2016. — С. 3073–3082. — DOI: 10.1109/CVPR.2016.335.

128. An Intriguing Failing of Convolutional Neural Networks and the CoordConv Solution / R. Liu [та ін.] // Proceedings of the 32nd International Conference on Neural Information Processing Systems. — Montreal, Canada : Curran Associates Inc., 2018. — С. 9628–9639. — (NIPS'18).

129. Fast Geometrically-Perturbed Adversarial Faces / A. Dabouei [та ін.] // 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). — 2019. — С. 1979–1988. — DOI: 10.1109/WACV.2019.00215.

130. *Zhang R.* Making Convolutional Networks Shift-Invariant Again // Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA. Т. 97. — PMLR, 2019. — С. 7324–7334. — (Proceedings of Machine Learning Research). — URL: http://proceedings.mlr.press/v97/zhang19a.html.

131. Large Deformation Diffeomorphic Metric Curve Mapping / J. A. Glaunès [та ін.] // Int. J. Comput. Vis. — 2008. — Т. 80, № 3. — С. 317–336. — DOI: 10.1007/s11263-008-0141-9.

132. *Joshi S. C.*, *Miller M. I.* Landmark matching via large deformation diffeomorphisms // IEEE Trans. Image Process. — 2000. — Т. 9, № 8. — С. 1357–1370. — DOI: 10.1109/83.855431.

133. Attention is All you Need / A. Vaswani [та ін.] // Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. — 2017. — С. 5998–6008. — URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

134. *Kazemi V.*, *Sullivan J.* One Millisecond Face Alignment with an Ensemble of Regression Trees // Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. — USA : IEEE Computer Society, 2014. — С. 1867–1874. — (CVPR '14). — ISBN 9781479951185. — DOI: 10.1109/CVPR.2014.241.

135. Densely Connected Convolutional Networks / G. Huang [та ін.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. — IEEE Computer Society, 2017. — С. 2261–2269. — DOI: 10.1109/CVPR.2017.243.

136. *Zagoruyko S.*, *Komodakis N.* Wide Residual Networks // Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016. — BMVA Press, 2016. — URL: http://www.bmva.org/bmvc/2016/papers/paper087/index.html.

137. *Finn C.*, *Abbeel P.*, *Levine S.* Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks // Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. T. 70. — PMLR, 2017. — C. 1126–1135. — (Proceedings of Machine Learning Research). — URL: http://proceedings.mlr.press/v70/finn17a.html.

138. *Yin W.* Meta-learning for Few-shot Natural Language Processing: A Survey // CoRR. — 2020. — T. abs/2007.09604. — arXiv: 2007.09604. — URL: https://arxiv.org/abs/2007.09604.

139. Generalizing from a Few Examples: A Survey on Few-shot Learning / Y. Wang [та ін.] // ACM Comput. Surv. — 2020. — Т. 53, № 3. — 63:1–63:34. — DOI: 10.1145/3386252.

140. *Guo Y.*, *Zhang L.* One-shot Face Recognition by Promoting Underrepresented Classes // CoRR. — 2017. — T. abs/1707.05574. — arXiv: 1707.05574. — URL: http://arxiv.org/abs/1707.05574.

141. *Weng L.* Meta-Learning: Learning to Learn Fast // lilianweng.github.io. — 2018. — URL: https://lilianweng.github.io/posts/2018-11-30-meta-learning/.

142. *Koch G.*, *Zemel R.*, *Salakhutdinov R.* Siamese Neural Networks for One-Shot Image Recognition // ICML Deep Learning Workshop. T. 2. — Lille, 2015.

143. Matching Networks for One Shot Learning / O. Vinyals [та ін.] // Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. — 2016. — C. 3630–3638. — URL:

https://proceedings.neurips.cc/paper/2016/hash/90e1357833654983612fb05e3ec9148c-Abstract.html.

144. *Snell J.*, *Swersky K.*, *Zemel R. S.* Prototypical Networks for Few-shot Learning // Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. — 2017. — С. 4077–4087. — URL: https://proceedings.neurips.cc/paper/2017/hash/cb8da6767461f2812ae4290eac7cbc42-Abstract.html.

145. Meta-Learning with Memory-Augmented Neural Networks / A. Santoro [та ін.] // Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016. Т. 48. — JMLR.org, 2016. — С. 1842–1850. — (JMLR Workshop and Conference Proceedings). — URL: http://proceedings.mlr.press/v48/santoro16.html.

146. *Lake B. M.*, *Salakhutdinov R.*, *Tenenbaum J. B.* Human-Level Concept Learning through Probabilistic Program Induction // Science. — 2015. — Т. 350, № 6266. — С. 1332–1338. — DOI: 10.1126/science.aab3050.

147. *Ravi S.*, *Larochelle H.* Optimization as a Model for Few-Shot Learning // 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. — OpenReview.net, 2017. — URL: https://openreview.net/forum?id=rJY0-Kcll.

148. *Nichol A.*, *Achiam J.*, *Schulman J.* On First-Order Meta-Learning Algorithms // CoRR. — 2018. — Т. abs/1803.02999. — arXiv: 1803.02999. — URL: http://arxiv.org/abs/1803.02999.

149. *Antoniou A.*, *Edwards H.*, *Storkey A. J.* How to train your MAML // 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. — OpenReview.net, 2019. — URL: https://openreview.net/forum?id=HJGven05Y7.

150. Meta-SGD: Learning to Learn Quickly for Few Shot Learning / Z. Li [та ін.] // CoRR. — 2017. — Т. abs/1707.09835. — arXiv: 1707.09835. — URL: http://arxiv.org/abs/1707.09835.

151. Meta-Learning with Implicit Gradients / A. Rajeswaran [та ін.] // Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. — 2019. — С. 113–124. — URL: https://proceedings.neurips.cc/paper/2019/hash/072b030ba126b2f4b2374f342be9ed44-Abstract.html.

152. *Goodfellow I.*, *Bengio Y.*, *Courville A.* Deep Learning. — MIT Press, 2016. — http://www.deeplearningbook.org.

153. *Deng L.* The mnist database of handwritten digit images for machine learning research // IEEE Signal Processing Magazine. — 2012. — Т. 29, № 6. — С. 141–142.

154. *Krizhevsky A.* Learning Multiple Layers of Features from Tiny Images : тех. звіт. / University of Toronto. — 2009.

155. Aggregated Residual Transformations for Deep Neural Networks / S. Xie [та ін.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. — IEEE Computer Society, 2017. — С. 5987–5995. — DOI: 10.1109/CVPR.2017.634. — URL: https://doi.org/10.1109/CVPR.2017.634.

156. *Müller S. G.*, *Hutter F.* TrivialAugment: Tuning-free Yet State-of-the-Art Data Augmentation // 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. — IEEE, 2021. — С. 754–762. — DOI: 10.1109/ICCV48922.2021.00081. — URL: https://doi.org/10.1109/ICCV48922.2021.00081.

157. CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features / S. Yun [та ін.] // 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. — IEEE, 2019. — С. 6022–6031. — DOI: 10.1109/ICCV.2019.00612. — URL: https://doi.org/10.1109/ICCV.2019.00612.

158. mixup: Beyond Empirical Risk Minimization / H. Zhang [та ін.] // 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. — OpenReview.net, 2018. — URL: https://openreview.net/forum?id=r1Ddp1-Rb.

159. *Loshchilov I.*, *Hutter F.* SGDR: Stochastic Gradient Descent with Warm Restarts // 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. — OpenReview.net, 2017. — URL: https://openreview.net/forum?id=Skq89Scxx.

160. Models and pre-trained weights. — 2023. — URL: https://pytorch.org/vision/stable/models.html (дата звернення: 03.05.2023).

161. PyTorch: An Imperative Style, High-Performance Deep Learning Library / A. Paszke [та ін.] // Advances in Neural Information Processing Systems 32. — Curran Associates, Inc., 2019. — С. 8024–8035. — URL: http:

//papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

162. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python / P. Virtanen [та ін.] // Nature Methods. — 2020. — Т. 17. — С. 261–272. — DOI: 10.1038/s41592-019-0686-2.

163. Algorithms for Hyper-Parameter Optimization / J. Bergstra [та ін.] // Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain. — 2011. — С. 2546–2554. — URL: https://proceedings.neurips.cc/paper/2011/hash/86e8f7ab32cfd12577bc2619bc635690-Abstract.html.

164. Optuna: A Next-generation Hyperparameter Optimization Framework / T. Akiba [та ін.] // Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019. — ACM, 2019. — С. 2623–2631. — DOI: 10.1145/3292500.3330701. — URL: https://doi.org/10.1145/3292500.3330701.

165. Dual Spoof Disentanglement Generation for Face Anti-spoofing with Depth Uncertainty Learning / H. Wu [та ін.] // CoRR. — 2021. — Т. abs/2112.00568. — arXiv: 2112.00568. — URL: https://arxiv.org/abs/2112.00568.

166. Biometric Presentation Attack Detection: Beyond the Visible Spectrum / R. Tolosana [та ін.] // IEEE Trans. Inf. Forensics Secur. — 2020. — Т. 15. — С. 1261–1275. — DOI: 10.1109/TIFS.2019.2934867.

167. *Kingma D. P.*, *Ba J.* Adam: A Method for Stochastic Optimization // 3rd International Conference on Learning Representations, ICLR 2015, San

Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. — 2015. — URL: http://arxiv.org/abs/1412.6980.

168. *Hardt D.* The OAuth 2.0 Authorization Framework. — 10.2012. — DOI: 10.17487/RFC6749. — URL: https://www.rfc-editor.org/info/rfc6749. RFC 6749.

169. ASP.NET documentation. — 2023. — URL: https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-7.0 (дата звернення: 07.05.2023).

170. PostgreSQL: The World's Most Advanced Open Source Relational Database. — 2023. — URL: https://www.postgresql.org/ (дата звернення: 07.05.2023).

171. Docker: Accelerated, Containerized Application Development. — 2023. — URL: https://www.docker.com/ (дата звернення: 07.05.2023).

172. Windows UI Library in the Windows App SDK (WinUI 3). — 2023. — URL: https://learn.microsoft.com/en-us/windows/apps/winui/winui3/ (дата звернення: 07.05.2023).

173. Xamarin.Forms documentation. — 2023. — URL: https://learn.microsoft.com/en-us/xamarin/xamarin-forms/ (дата звернення: 07.05.2023).

174. Jetpack Compose UI App Development Toolkit. — 2023. — URL: https://developer.android.com/jetpack/compose (дата звернення: 07.05.2023).

175. NFC basics. — 2020. — URL: https://developer.android.com/guide/topics/connectivity/nfc/nfc (дата звернення: 12.10.2020).

176. What is GUID? — 2020. — URL: http://guid.one/guid (дата звернення: 24.11.2020).

177. Allowing Apps and Websites to Link to Your Content | Apple Developer Documentation. — 2020. — URL: https://developer.apple.com/documentation/xcode/allowing_apps_and_websites_to_link_to_your_content (дата зверн. 23.10.2020).

178. Core NFC | Apple Developer Documentation. — 2020. — URL: https://developer.apple.com/documentation/corenfc (дата зверн. 23.10.2020).

179. Adding Support for Background Tag Reading | Apple Developer Documentation. — 2020. — URL: https://developer.apple.com/documentation/corenfc/adding_support_for_background_tag_reading (дата зверн. 23.10.2020).

180. *Ojala T.*, *Pietikäinen M.*, *Mäenpää T.* Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns // IEEE Trans. Pattern Anal. Mach. Intell. — 2002. — Т. 24, № 7. — С. 971–987. — DOI: 10.1109/TPAMI.2002.1017623. — URL: https://doi.org/10.1109/TPAMI.2002.1017623.

181. *Bradski G.* The OpenCV Library // Dr. Dobb's Journal of Software Tools. — 2000.

182. *King D. E.* Dlib-Ml: A Machine Learning Toolkit // J. Mach. Learn. Res. — 2009. — Груд. — Т. 10. — С. 1755–1758. — ISSN 1532-4435.

183. Microsoft COCO: Common Objects in Context / T. Lin [та ін.] // Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V. Т. 8693 / за ред. D. J. Fleet [та ін.]. — Springer, 2014. — С. 740–755. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-319-10602-1_48.

184. The Pascal Visual Object Classes (VOC) Challenge / M. Everingham [та ін.] // Int. J. Comput. Vis. — 2010. — Т. 88, № 2. — С. 303–338. — DOI: 10.1007/s11263-009-0275-4. — URL: https://doi.org/10.1007/s11263-009-0275-4.

185. The Cityscapes Dataset for Semantic Urban Scene Understanding / M. Cordts [та ін.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. — IEEE Computer Society, 2016. — С. 3213–3223. — DOI: 10.1109/CVPR.2016.350.

186. *Khabarlak K.* Face Detection on Mobile: Five Implementations and Analysis // CoRR. — 2022. — Т. abs/2205.05572. — DOI: 10.48550/arXiv.2205.05572. — arXiv: 2205.05572.

187. *Milletari F.*, *Navab N.*, *Ahmadi S.* V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation // Fourth International Conference on 3D Vision, 3DV 2016, Stanford, CA, USA, October 25-28, 2016. — IEEE Computer Society, 2016. — С. 565–571. — DOI: 10.1109/3DV.2016.79. — URL: https://doi.org/10.1109/3DV.2016.79.

188. Meta-learning with differentiable closed-form solvers / L. Bertinetto [та ін.] // 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. — OpenReview.net, 2019. — URL: https://openreview.net/forum?id=HyxnZh0ct7.

189. *Zeiler M. D.*, *Fergus R.* Visualizing and Understanding Convolutional Networks // Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I. Т. 8689. — Springer, 2014. — С. 818–833. — (Lecture Notes in Computer Science). — DOI: 10.1007/978-3-319-10590-1_53.

ДОДАТОК А
СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні наукові результати дисертації:

Публікації у фахових виданнях України:

1. *Khabarlak K. S.*, *Koriashkina L. S.* Mobile Access Control System Based on RFID Tags and Facial Information // Bulletin of National Technical University ”KhPI”. Series: System Analysis, Control and Information Technologies. 2020. Т. 2, № 4. С. 69–74. DOI: 10.20998/2079-0023.2020.02.12. URL: http://samit.khpi.edu.ua/article/view/2079-0023.2020.02.12 **(Фаховий (категорія Б))**.
2. *Хабарлак К. С.* Особливості роботи методів пошуку облич на мобільних пристроях // System Technologies. 2021. Т. 6, № 137. С. 34–45. DOI: 10.34185/1562-9945-6-137-2021-04. URL: https://journals.nmetau.edu.ua/index.php/st/issue/view/118/97 **(Фаховий (категорія Б))**.
3. *Khabarlak K.* Post-Train Adaptive U-Net for Image Segmentation // Information Technology: Computer Science, Software Engineering and Cyber Security. 2022. № 2. С. 73–78. DOI: 10.32782/IT/2022-2-8. URL: https://doi.org/10.32782/IT/2022-2-8 **(Фаховий (категорія Б))**.

Публікації у виданнях, які проіндексовані у міжнародних наукометричних базах Web of Science та Scopus:

1. *Khabarlak K.*, *Koriashkina L.* Fast Facial Landmark Detection and Applications: A Survey // Journal of Computer Science and Technology.

2022. Квіт. Т. 22, № 1. С. 12–41. DOI: 10.24215/16666038.22.e02. URL: https://journal.info.unlp.edu.ar/JCST/article/view/1972 **(Scopus, Web of Science, закордонний журнал)**.

2. *Khabarlak K. S.* Faster Optimization-Based Meta-Learning Adaptation Phase // Radio Electronics, Computer Science, Control. 2022. Квіт. № 1. С. 82–92. DOI: 10.15588/1607-3274-2022-1-10. URL: http://ric.zntu.edu.ua/article/view/254615 **(Web of Science, Фаховий (категорія А))**.
3. *Khabarlak K. S.*, *Koriashkina L. S.* Scoping Adversarial Attack for Improving Its Quality // Radio Electronics, Computer Science, Control. 2019. Трав. № 2. С. 108–118. DOI: 10.15588/1607-3274-2019-2-12. URL: http://ric.zntu.edu.ua/article/view/178284 **(Web of Science, Фаховий (категорія А))**.

Наукові праці, які засвідчують апробацію матеріалів дисертації:

Авторське свідоцтво:

1. Комп'ютерна програма «Мобільна нейромережева система пошуку облич із анті-спуфінгом» : авт. свід. України №110917 / К. С. Хабарлак. 11.01.2022

Матеріали конференцій та тези доповідей:

1. *Khabarlak K.* Post-Train Adaptive MobileNet for Fast Anti-Spoofing // Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security, Khmelnytskyi, Ukraine, March 23–25. Т. 3156. CEUR-WS.org, 2022. С. 44–53. (CEUR Workshop Proceedings). URL: http://ceur-ws.org/Vol-3156/keynote5.pdf **(Scopus)**

2. *Хабарлак К. С.*, *Коряшкіна Л. С.* Деякі особливості гіперпараметрів глибоких нейронних мереж // III Всеукраїнська Інтернет-конференція здобувачів вищої освіти і молодих учених «Інформаційні технології: теорія і практика». Харків, 03.2020. С. 98–99
3. *Khabarlak K.*, *Koriashkina L.* Top image classification accuracy through hyperparameter search // 15th International Forum for Students and Young Researchers “Widening our horizons”. Dnipro, 05.2020. С. 271–274
4. *Khabarlak K.* Mobile Application for RFID Access Control System // V міжнародна науково-практична конференція «Прикладні науково-технічні дослідження». Івано-Франківськ, 04.2021. С. 99–100
5. *Хабарлак К. С.* Анти-спуфінг для системи контролю доступу із RFID мітками // Збірник матеріалів III Всеукраїнської конференції «Теоретико-практичні проблеми використання математичних методів і комп’ютерно-орієнтованих технологій в освіті та науці». Київ, 04.2021. С. 141–142
6. *Хабарлак К. С.* On Face Detection and Anti-Spoofing in Mobile Access Control // Тези доповідей VIII Міжнародної науково-практичної конференції «Інформаційні технології в освіті, науці і виробництві (ІТОНВ-2021)». Луцьк, 05.2021. С. 181–183
7. *Хабарлак К. С.*, *Коряшкіна Л. С.* Тестування швидкості виконання алгоритмів пошуку обличчь для системи контролю доступу // VII Всеукраїнська науково-технічна конференція молодих учених, аспірантів та студентів «Автоматизація, контроль та управління: пошук ідей та рішень» (АКУ-2021). Покровськ, 05.2021. С. 60–61
8. *Хабарлак К. С.* Про адаптацію мета-навчання нейронних мереж // Матеріали міжнародної конференції II International Scientific Symposium “Intelligent Solutions-S”. Ужгород, 09.2021. С. 81
9. *Хабарлак К. С.* Адаптація мета-навчання на частковому шаблоні //

Матеріали VII міжнародної науково-технічної конференції Комп'ютерне моделювання та оптимізація складних систем. Дніпро, 11.2021. С. 121–122

10. *Хабарлак К. С.* Аналіз шаблонів адаптації мета-навчання // Матеріали IX Всеукраїнської науково-технічної конференції студентів, аспірантів та молодих вчених «Молодь: наука та інновації». Дніпро, 11.2021. С. 328–329
11. *Хабарлак К. С.* Прискорене навчання нейронної мережі за декількома прикладами // XVI Міжнародна конференція з проблем використання інформаційних технологій в освіті, науці та промисловості. Дніпро, 12.2021. С. 62–64
12. *Хабарлак К. С.* Зміна архітектури нейронної мережі після навчання // Тези дев'ятої міжнародної науково-технічної конференції Інформатика, управління та штучний інтелект. Харків – Краматорськ, 05.2022. С. 135
13. *Хабарлак К. С.* Нейромережний пошук об'єктів на мобільних пристроях // Матеріали XII Всеукраїнської науково-технічної конференції студентів, аспірантів та молодих вчених «Наукова весна». Дніпро, 05.2022. С. 171
14. *Хабарлак К. С.* Адаптивна після навчання нейронна мережа // Тези V Всеукраїнської Інтернет-конференції здобувачів вищої освіти і молодих учених Інформаційні технології: теорія і практика. Запоріжжя, 06.2022. С. 20–21
15. *Хабарлак К. С.* Нейро-мережева система класифікації із конфігурацією після навчання // Матеріали X Міжнародної науково-технічної конференції студентів, аспірантів і молодих вчених «Молодь: наука та інновації». Дніпро, 11.2022. С. 383
16. *Khabarlak K.* Semantic segmentation with Post-Train Adaptive Neural Network // Тези XI міжнародної науково-практичної конференції

«Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій». Запоріжжя, 12.2022. С. 124–125

17. *Хабарлак К. С.* Конфігурація після навчання нейронної мережі для сегментації зображень // Матеріали XIII Міжнародної науково-технічної конференції аспірантів та молодих вчених «Наукова весна». Дніпро, 03.2023. С. 194–195
18. *Хабарлак К. С.* Проблеми нейронних мереж для розпізнавання на пристроях із різними обчислювальними можливостями // Тези VI Всеукраїнської науково-практичної Інтернет-конференції здобувачів вищої освіти і молодих учених Інформаційні технології: теорія і практика. Харків, 03.2023. С. 101–102

ДОДАТОК Б

АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

«ЗАТВЕРДЖЕНО»
Проректор з науково-педагогічної
роботи Національного технічного
університету «Дніпровська політехніка»
_____ Артем ПАВЛИЧЕНКО
«03» липня 2023р.

АКТ
про впровадження результатів дисертаційної роботи на здобуття наукового ступеня доктора філософії в навчальний процес Національного технічного університету «Дніпровська політехніка»

Результати дисертаційної роботи Хабарлака Костянтина Сергійовича «Методи класифікації та сегментації зображень на основі змінюваних згорткових мереж» використовуються у Національному технічному університеті «Дніпровська політехніка» в навчальному процесі підготовки студентів за напрямами 124 Системний аналіз і 122 Комп'ютерні науки при викладанні наступних дисциплін:

- Самонавчання складних систем, розділи: «Нейронні мережі. Основні поняття», «Глибоке навчання згорткових нейронних мереж», «Розробка рекурентної мережі», «Просунуте машинне навчання»;
- Methods and Systems of Artificial Intelligence (для іноземних студентів), розділи: "Introduction into Artificial Intelligence Systems", "Neural Networks and Computer Vision", "Neural Networks and Text Processing", "Advanced Topics in Neural Networks", "Implementing a Fully-Connected Neural Network".

На основі дисертаційних досліджень підготовлено лекційні матеріали та розроблені завдання до лабораторних робіт.

Декан факультету інформаційних технологій, к.т.н., доцент — Ірина УДОВИК

Зав. кафедрою системного аналізу та управління, к.т.н., доцент — Тимур ЖЕЛДАК

ДОДАТОК В
АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ПРИДНІПРОВСЬКИЙ НАУКОВИЙ ЦЕНТР

49005, м. Дніпро, а/с 484 Телефон/факс: +38 (067) 257-50-44
e-mail: office.pse@nas.gov.ua; http://www.nas.gov.ua/pne код ОКПО 01209713

вих. № 23/30 від 06.04.2023

Щодо впровадження результатів
дисертаційного дослідження

АКТ

про впровадження результатів дисертаційної роботи

на здобуття наукового ступеня доктора філософії

Хабарлака Костянтина Сергійовича

Результати досліджень, які представлені в дисертаційній робот Хабарлака Костянтина Сергійовича можуть бути використані задля підвищення безпеки дорожнього руху в Дніпропетровській області або в Україні за допомогою розробленої системи контролю стану водія.

Велика кількість аварій відбувається через сонних, втомлених водіїв або через відволікання водія від дороги. Розроблена система на основі змінюваної згорткової мережі дозволяє ефективно та в реальному часі відстежувати такі випадки та змушувати водія зосередитись на дорозі або зупинити транспортний засіб. Для виконання нейронної мережі використовується крайовий пристрій або мобільний телефон водія для обробки інформації. Система може бути вбудована для контролю водіїв громадського транспорту, таксі, вантажних транспортних

засобів та каршерінгу задля зменшення аварійності. Обробка проводиться прямо на пристрої, що дозволяє реалізувати таку систему в умовах поганого доступу до мережі Інтернет та уникаючи надмірного навантаження на центральний сервер.

Акт впровадження результатів дисертаційних досліджень видано для представлення у спеціалізовану вчену раду.

Директор ПНЦ НАН України і МОН України
член-кореспондент НАН України **Борис БЛЮСС**

Учений секретар ПНЦ НАН України і МОН України
д-р техн. наук, с.н.с. **Сергій ДЗЮБА**

ДОДАТОК Г
АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
ІНСТИТУТ ГЕОТЕХНІЧНОЇ МЕХАНІКИ ім. М.С. Полякова

вул. Сімферопольська, 2А
м. Дніпро, 49005
код ЄДРПОУ 05411357

E-mail: office.igtm@nas.gov.ua
igtmnanu@ukr.net
тел. 38 050 5317375

«15» березня 2023 року
вих. №311-22/11-1-4

Щодо впровадження результатів дисертаційного дослідження

АКТ

про впровадження результатів дисертаційної роботи
на здобуття наукового ступеня доктора філософії
Хабарлака Костянтина Сергійовича

Результати кандидатської дисертації, які отримані асистентом кафедри системного аналізу та управління НТУ «Дніпровська політехніка» Хабарлаком Костянтином Сергійовичем при виконанні дисертаційного дослідження є актуальними, мають наукове і практичне значення для виробничих підприємств, зокрема підприємств гірничо-металургійної галузі завдяки розробленій системі контролю доступу із RFID мітками та підсистемою антиспуфінгу.

Основним компонентом представленої в дисертаційному досліджені Хабарлака К.С. системи контролю доступу із RFID мітками є мобільний застосунок, який 1) зчитує інформацію про технологічне обладнання або розумні двері з мітки; 2) проводить ідентифікацію користувача за його обліковим записом та обличчям. Після цього, надаючи інформацію про перевірку до серверу,

система вирішує дозволити доступ до технологічного обладнання або дверей чи ні. Додатково система може автоматично відстежувати носіння предметів індивідуального захисту передбачених для користування обладнанням, таких як захисний шолом, окуляри, протигаз тощо, та заборонити роботу без них. Всі спроби доступу відстежуються та представлені на панелі моніторингу на ПК особи, відповідальної за безпеку на підприємстві. Окремого відзначення заслуговує те, що перевірка особистості користувача та носіння захисного обладнання виконується за допомогою представленої в роботі Хабарлака К.С. нейронної мережі із РТА блоками прямо на мобільному пристрої.

Акт впровадження результатів дисертаційних досліджень видано для представлення у спеціалізовану вчену раду.

Заступник директора
з наукової роботи
чл.-кор. НАН України **Олександр КРУКОВСЬКИЙ**

Зав. відділу
управління динамічними проявами
гірничого тиску
д-р техн. наук, проф. **Сергій МІНЄЄВ**