

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України

Кваліфікаційна наукова
праця на правах рукопису

Переверзєв Олексій Андрійович

УДК 621.391.83

ДИСЕРТАЦІЯ
ВІДКРИТІ НЕЙРОННІ МЕРЕЖІ В АЛГОРИТМАХ РОЗПІЗНАВАННЯ
ВІЗУАЛЬНОЇ ІНФОРМАЦІЇ ДЛЯ ЕЛЕКТРОННИХ СИСТЕМ БЕЗПЕКИ

17 – Електроніка та телекомунікації

171 – Електроніка

Подається на здобуття наукового ступеня доктора філософії.

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

_____ / Переверзєв О.А.

Науковий керівник: Трапезон Кирило Олександрович, кандидат технічних наук,
доцент

Київ - 2025

АНОТАЦІЯ

Переверзєв О.А. Відкриті нейронні мережі в алгоритмах розпізнавання візуальної інформації для електронних систем безпеки. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії у галузі знань 17 – Електроніка та телекомунікації за спеціальністю 171 – Електроніка. – Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Міністерство освіти і науки України, Київ, 2025.

Дисертація присвячена аналізу та вдосконаленню сучасних алгоритмів розпізнавання візуальної інформації, що мають важливе практичне значення для забезпечення ефективності функціонування електронних систем безпеки. У дослідженні зосереджено увагу на інтеграції та удосконаленні алгоритмів глибинного навчання, серед яких обрано модель YOLOv7, спрямованих на розв'язання проблем, пов'язаних із низькою чіткістю зображення, шумом, оклюзією та змінним освітленням.

Зміст дисертаційного дослідження викладено в трьох розділах, де представлено та обґрунтовано основні отримані результати роботи.

Актуальність дисертаційної роботи обґрунтовано у вступі, де сформульовано мету та задачі дослідження, описано методи дослідження, надано інформацію про наукову новизну та практичне значення одержаних результатів.

Перший розділ дисертаційного дослідження присвячено аналізу існуючих алгоритмів розпізнавання візуальної інформації, що використовуються в електронних системах безпеки. Розглянуто основні принципи та класифікацію алгоритмів виявлення і відстеження об'єктів, їх застосування у відеоспостереженні, військових системах та транспортних технологіях.

Проаналізовано перспективні методи розпізнавання об'єктів, зокрема CAMShift, Optical Flow, для яких визначено переваги та недоліки при зміні

умов освітлення, динаміці сцени та наявності шуму. Додатково досліджено підходи до розпізнавання тексту на зображеннях.

У розділі також проведено порівняння традиційних алгоритмів із методами глибокого навчання. Визначено, що нейронні мережі демонструють вищу ефективність у порівнянні з традиційними способами, при роботі з візуальною інформацією в умовах часткового перекриття, змінного освітлення та складного фону, що підтверджує їхню доцільність для впровадження у системи відеоспостереження та безпеки.

Отримані результати демонструють переваги використання адаптивних методів обробки відеопотоку та підходів до динамічного налаштування параметрів алгоритмів відстеження. Це дозволяє значно покращити продуктивність електронних систем безпеки в реальних умовах експлуатації та підвищити їх стійкість до зовнішніх перешкод.

Другий розділ дисертаційного дослідження присвячено експериментальному аналізу алгоритмів виявлення, відстеження та обробки візуальної інформації у системах безпеки. Основний акцент зроблено на оцінці ефективності двох найбільш поширених алгоритмів CAMShift та Optical Flow для відстеження об'єктів у відеопотоці, а також на порівнянні їхньої продуктивності в умовах змінного освітлення, часткової оклюзії та шумових перешкод.

Додатково у розділі розглянуто методи зменшення шуму на зображеннях шляхом використання згорткових нейронних мереж. Проведено аналіз підходів до фільтрації яскравісного шуму, що виникає внаслідок роботи світлочутливої матриці камери, та впливу при фіксації зображення зовнішніх факторів (туман, затемнення, забруднення камери).

Також досліджено підходи до розпізнавання тексту на зображеннях за допомогою одного з виду вейвлет-перетворень, що дозволяє зменшити вплив фонових завад та підвищити точність обробки зображень у складних умовах.

Важливу увагу приділено експериментальному порівнянню різних модифікацій YOLOv7, оцінці впливу змін у структурі нейромережі на точність

виявлення та швидкість обробки відеопотоку. Проаналізовано ефективність додавання каскадних шарів та механізму уваги для покращення роботи нейромереж у складних сценах.

У третьому розділі роботи зосереджено увагу на практичній реалізації та перевірці алгоритмів оцінки об'єктів на поверхні зображення з камери відеоспостереження на основі застосування моделі YOLO. Основна увага прикута дослідженню адаптації нейронних мереж до реальних умов експлуатації, створенню спеціалізованих датасетів і впровадженню моделей у системи відеоспостереження.

Особливістю третього розділу є перевірка моделі YOLOv7 та розроблення її модифікації для використання в системі відеоспостереження за складних оточуючих умов фіксації. Проведено серію експериментів, які включали аналіз відеопотоку в умовах різної прозорості середовища (імітація туману, забруднення камери), часткової оклюзії об'єктів і повної відсутності освітлення а також особливості роботи в інфрачервоному режимі.

У дисертації представлено наступні наукові результати:

1. Вперше розроблено модифіковану систему розпізнавання зображень, яка дозволяє ідентифікувати об'єкти холодної зброї в інфрачервоному режимі та в умовах низької видимості.

2. Удосконалено роботу нейронної мережі згорткового типу, яка дозволяє при обробленні зображень зменшувати адитивний яскравісний шум зображення у випадку, коли сам рівень шуму може змінюватись в при аналізі секвенції зображень.

3. Удосконалено оптичну систему розпізнавання тексту, яка характеризується стійкістю до заважаючих фонових елементів у формі кривих Без'є.

4. Удосконалено архітектуру моделі YOLO V7 за рахунок додавання капсульного шару, внаслідок чого, знайдено вирішення проблеми обмеженого виявлення об'єктів за відстанню, у складних сценах із частковим перекриттям

або зміною форми останніх а також в умовах зменшеної видимості об'єктів зображення за наявності або відсутності освітлення.

Практичне значення отриманих результатів полягає у наступному.

1. Визначені рекомендації щодо оцінки моделей на основі YOLOv7 дозволяють удосконалити їхню роботу в умовах низького освітлення, зменшеної чіткості об'єктів та часткового перекриття, що є актуальним для систем, які працюють у складних умовах.

2. Отримані результати можуть бути інтегровані у прикладні рішення для автоматизованого виявлення зброї на інфрачервоних та оптичних зображеннях, що сприятиме у подальшому підвищенню рівня безпеки в громадських та стратегічно важливих об'єктах.

3. Запропоновані підходи до адаптації алгоритмів розпізнавання об'єктів можуть бути використані у військових системах, транспортних технологіях, інтелектуальних системах управління дорожнім рухом та інших галузях, де важливе своєчасне і точне розпізнавання об'єктів.

4. Практичні результати можуть знайти своє застосування у розробці програмного забезпечення для аналізу відеопотоку, орієнтованого на роботу з інфрачервоними камерами, що забезпечить можливість ефективного моніторингу в умовах поганої видимості.

Ключові слова: згортка, мультимедійна інформація, мережа, оцінка ефективності, контент, безпека, модель, шуми, моделювання, Інтернет речей, рівень сигналу, IoT, розповсюдження сигналу, оцінка, тестовий сигнал, зображення.

ABSTRACT

Pereverziev O.A. Open neural networks in visual information recognition algorithms for electronic security systems. – A qualification scientific work as a manuscript.

Dissertation for the degree of Doctor of Philosophy in the field of knowledge 17 – Electronics and Telecommunications, specialty 171 – Electronics. – National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ministry of Education and Science of Ukraine, Kyiv, 2025.

The dissertation is devoted to the analysis and improvement of modern visual information recognition algorithms that are of great practical importance for ensuring the efficiency of electronic security systems. The research focuses on the integration of deep learning algorithms, such as YOLOv7, into image and video stream analysis processes, as well as on solving problems associated with low image clarity, noise, occlusion, and variable lighting.

The content of the dissertation research is outlined in three chapters, where the main results of the work are presented and substantiated.

The relevance of the dissertation is substantiated in the introduction, which formulates the purpose and objectives of the study, describes the research methods, and provides information on the scientific novelty and practical significance of the results.

The first chapter of the dissertation is devoted to the analysis of visual information recognition algorithms used in electronic security systems. The basic principles and classification of object detection and tracking algorithms, their application in video surveillance, military systems and transport technologies are considered.

Traditional methods of object recognition, such as CAMShift and Optical Flow, are analysed, as well as their advantages and disadvantages in changing lighting conditions, scene dynamics and noise. Additionally, approaches to text recognition in images are investigated.

The chapter also compares traditional algorithms with deep learning methods. It is determined that neural networks demonstrate higher efficiency when working in conditions of partial overlap, variable lighting and complex backgrounds, which confirms their feasibility for implementation in video surveillance and security systems.

The obtained results demonstrate the advantages of using adaptive video stream processing methods and approaches to dynamically adjusting the parameters of tracking algorithms. This can significantly improve the performance of electronic security systems in real-world conditions and increase their resistance to external interference.

The second chapter of the dissertation is devoted to the experimental analysis of algorithms for detecting, tracking and processing visual information in security systems. The main focus is on evaluating the effectiveness of CAMShift and Optical Flow algorithms for tracking objects in a video stream, as well as comparing their performance under conditions of variable lighting, partial occlusion, and noise interference.

Additionally, the chapter discusses methods for reducing noise in images using convolutional neural networks. The approaches to filtering luminance noise arising from poor video stream quality and the influence of external factors (fog, blackout, camera contamination) are analysed.

Approaches to text recognition in images using wavelet transforms are also investigated, which reduces the influence of background noise and increases the accuracy of image processing in difficult conditions.

Important attention is paid to the experimental comparison of various modifications of YOLOv7, assessing the impact of changes in the structure of the neural network on the detection accuracy and processing speed of the video stream. The effectiveness of adding cascade layers and an attention mechanism to improve neural network performance in complex scenes is analysed.

The third section of the paper focuses on the practical implementation and verification of algorithms for estimating objects on the surface of a video surveillance

camera image based on the YOLO model. The main focus is on the adaptation of neural networks to real operating conditions, the creation of specialised datasets and the implementation of models in video surveillance systems.

The third chapter is dedicated to the adaptation of software to work with the modified YOLOv7 model. The peculiarities of integrating a neural network into a video surveillance system are covered in detail.

The study also covers the analysis of methods for improving the accuracy of neural networks when working with a video stream in difficult conditions. Particular attention is paid to the adaptation of convolutional neural networks to reduce the additive luminance noise that occurs during the processing of dynamic video. Formation of specialised datasets for training YOLOv7, which take into account the factors of partial occlusion, dynamic background and variable image quality.

A series of experiments were conducted, including analysis of the video stream in conditions of different environmental transparency (fog simulation, camera contamination), partial occlusion of objects and complete absence of lighting, as well as features of work in the infrared mode.

The dissertation presents the following scientific results:

1. Probably for the first time, a modified image recognition system has been developed that allows the identification of weapons in infrared and in extremely poor visibility conditions.
2. The work of the convolutional neural network has been improved, which allows to reduce the additive luminance noise of the image in the case when the noise level itself can change during the analysis of the image sequence.
3. An improved optical text recognition system characterized by resistance to interfering background elements in the form of Bezier curves.
4. The architecture of the YOLO V7 model has been improved by adding a capsule layer after the neck module, which solved the problem of detecting objects in complex scenes with partial overlap or change in shape of the latter, as well as in conditions of reduced visibility of image objects in the presence and absence of lighting.

The practical significance of the obtained results includes the following:

1. The defined parameters and recommendations for the use of models based on YOLOv7 allow to improve their performance in low light conditions, reduced object clarity and partial overlap, which is relevant for systems operating in difficult conditions.

2. The obtained results can be integrated into applied solutions for automated weapon detection on infrared and optical images, which will help to increase the level of security in public and strategically important facilities.

3. The proposed approaches to the adaptation of object recognition algorithms can be used in military systems, transportation technologies, intelligent traffic management systems, and other areas where timely and accurate object recognition is important.

4. Practical results can be applied in the development of video stream analysis software focused on working with infrared cameras, which will provide the possibility of effective monitoring in low visibility conditions.

Keywords: convolution, multimedia information, network, performance evaluation, content, security, model, noise, modeling, Internet of Things, signal strength, IoT, signal propagation, evaluation, test signal, image.

Список публікацій здобувача

1. Переверзєв О.А., Трапезон К.О. Дослідження програмних алгоритмів відстеження руху об'єктів в електронних системах безпеки // Вчені записки Таврійського національного університету імені В.І.Вернадського. Серія: Технічні науки. Том 33(72), №6, 2022. С.288-293. DOI: <https://doi.org/10.32782/2663-5941/2022.6/47>

2. Переверзєв О., Трапезон К. Особливості використання нейронних мереж у задачах зменшення шуму цифрових зображень // “Вісник КрНУ імені Михайла Остроградського”. – 2023. - №3 (140). – С. 44-51. DOI: <https://doi.org/10.32782/1995-0519.2023.3.6> (ISSN 1995–0519 print; ISSN 2072-8263 online)

3. Переверзєв О., Трапезон К. Програмні підходи використання вейвлет-перетворення в задачах розпізнавання тексту із зображень // “Вісник КрНУ імені Михайла Остроградського”. – 2024. - №1 (144). – С. 266-272. DOI: <https://doi.org/10.32782/1995-0519.2024.1.35> (ISSN 1995–0519 print; ISSN 2072-8263 online).

4. Переверзєв О., Трапезон К. Принципи та підходи використання алгоритму YOLO v7 в системах виявлення об'єктів // “Вісник КрНУ імені Михайла Остроградського”. – 2024. - №4 (147). – С. 104-112. DOI: <https://doi.org/10.32782/1995-0519.2024.4.13> (ISSN 1995–0519 print; ISSN 2072-8263 online).

Доповіді на конференціях:

1. Переверзєв О., Трапезон К. Аналіз алгоритмів моніторингу об'єктів у системах безпеки // Матеріали ІХ Міжнародної науково-практичної конференції “Perspectives of Contemporary Science: Theory and Practice” (14–16 жовтня 2024 р., Львів, Україна). Львів, 2024.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	14
ВСТУП.....	15
1 ОГЛЯД АЛГОРИТМІВ РОЗПІЗНАВАННЯ ВІЗУАЛЬНОЇ ІНФОРМАЦІЇ ДЛЯ СИСТЕМ БЕЗПЕКИ.....	21
1.1 Основні принципи та класифікація алгоритмів виявлення і відстеження об'єктів.....	21
1.2 Традиційні методи: алгоритми CAMShift, Optical Flow та їх характеристика.....	23
1.3 Підходи розпізнавання тексту на зображеннях.....	26
1.4 Проблеми оклюзії, шуму та фону в задачах розпізнавання об'єктів.....	29
1.5 Порівняння нейронних мереж із традиційними методами обробки зображень.....	31
1.6 Модель YOLO для виявлення об'єктів на зображеннях.....	34
1.7 Прикладні задачі застосування нейронних мереж.....	38
1.7.1. Можливості нейронних мереж у системах відеоспостереження та контролю дорожнього трафіку.....	38
1.7.2 Особливості нейронних мереж у військових системах....	40
2 ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА ДОСЛІДЖЕННЯ.....	45
2.1 Оцінка якості відстеження об'єктів на основі алгоритмів CAMShift та Optical Flow.....	45
2.2 Алгоритм зменшення шуму на основі згорткових нейронних мереж.....	50
2.3 Підходи до застосування алгоритмів вейвлет-перетворення для аналізу тексту на зображеннях.....	58
2.4 Експериментальні дослідження різних модифікацій моделі YOLO v7.....	66
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ РОБОТИ АЛГОРИТМІВ ДЛЯ ЕЛЕКТРОННИХ СИСТЕМ БЕЗПЕКИ.....	74

3.1 Розробка електронної системи безпеки.....	75
3.2 Додаткові можливості електронної системи безпеки.....	80
3.3 Формування спеціалізованого датасету для навчання нейронної мережі.....	82
3.4 Адаптація програмного забезпечення для розпізнавання зброї	85
3.4.1 Обмеження класичної моделі YOLOV7.....	85
3.4.2 Процес навчання YOLO V7.....	87
3.5 Підключення обробки відеопотоку в реальному часі з використанням моделі YOLOV7.....	90
3.6 Логіка роботи системи обробки відеопотоку в електронній системі безпеки.....	91
3.7 Нові пропозиції в архітектурі моделі YOLO v7.....	93
3.8 Перевірка навченої моделі на предмет можливості детектування зброї при наявності освітлення.....	99
3.9 Перевірка моделі в умовах зменшеної чіткості при наявності освітлення (з імітацією несприятливих умов фіксації).....	103
3.10 Адаптація програмного забезпечення для розпізнавання зброї (ножів) з інфрачервоного зображення.....	106
3.11 Перевірка навченої моделі на предмет можливості детектування зброї при відсутності освітлення.....	110
3.12 Перевірка моделі в умовах зменшеної чіткості і при відсутності освітлення (імітація несприятливих умов).....	112
3.13 Аналіз отриманих результатів практичного експерименту.....	117
3.14 Оцінка ефективності модифікованої моделі YOLOv7.....	119
ВИСНОВКИ.....	123
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	126
ДОДАТОК А.....	137
ДОДАТОК Б.....	139

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

- ALPR – Automatic License Plate Recognition;
- CNN – Convolutional Neural Network;
- DWT – Discrete Wavelet Transform;
- FPS – Frames per second;
- GDDR6 – Graphics Double Data Rate 6 Synchronous Dynamic Random-Access Memory;
- GPU – Graphics processing unit;
- HDD – Hard disk drive;
- HDMI – High-Definition Multimedia Interface;
- IOU – Intersection Over Union;
- IP65 – Ingress protection code;
- NMS – Non-Maximum Suppression;
- OCR – Optical Character Recognition;
- PGI – Programmable Gradient Information;
- PSNR – Peak signal-to-noise ratio;
- RNN – Recurrent neural network;
- SSD – Single Shot Multibox Detector;
- SSIM – Structural similarity index measure;
- YOLO – You Only Look Once;
- RTSP – Real Time Streaming Protocol.

ВСТУП

Актуальність теми. У сучасному світі гостро постає питання забезпечення ефективного функціонування електронних систем безпеки, які здатні швидко та точно аналізувати навколишнє середовище з визначенням об'єктів, їх класифікації за основними ознаками в умовах реального часу. Зростаючий рівень загроз, включаючи злочинність, терористичні акти та техногенні ризики, вимагає впровадження у технологічну структуру мегаполісу автоматизованих систем розпізнавання об'єктів, які функціонують і дозволяють знизити залежність від людського фактору та підвищують рівень оперативності реагування.

Методи аналізу візуальних даних, такі як CAMShift, Optical Flow та інші, які використовуються сьогодні при аналізі зображень з камер, натомість мають низку обмежень, особливо в умовах складного фону, змін освітлення або оклюзії об'єктів [1]. Ці обмеження можуть стати перешкодою для забезпечення сучасних потреб у високоточних системах безпеки, насамперед, коли необхідно швидко визначити на динамічному зображенні з камери об'єкт, його тип та приналежність до певного класу. Виходом з цього, можуть бути підходи, які пов'язані з застосуванням методів глибинного навчання на основі впровадження згорткових нейронних мереж (Convolutional Neural Networks, CNN), які характеризуються тим, що можуть автоматично виділяти ключові ознаки об'єктів.

Так, особливої актуальності набуває практичне використання алгоритмів роботи нейронних мереж у системах відеоспостереження, контролю доступу та моніторингу критичних об'єктів. Однак існує ряд обмежень, які необхідно враховувати, таких як необхідність адаптації алгоритмів для роботи з низькоякісними зображеннями, шумами та умовами обмеженої видимості самих об'єктів на зображенні. Подолання цих проблем відкриває можливості для підвищення точності роботи і надійності систем розпізнавання об'єктів при аналізі динамічних зображень, які фіксуються, наприклад, камерами систем безпеки.

Враховуючи зростаючий попит на інтеграцію інноваційних технологій у сфері безпеки, а також значущість розробки стійких до складних умов алгоритмів, тема використання нейронних мереж для аналізу візуальної інформації з електронних систем безпеки є надзвичайно актуальною як у науковому, так і у практичному контексті.

Питаннями застосування різних алгоритмів для вирішення задач оброблення відеоінформації у частині розпізнавання об'єктів займаються такі вітчизняні та іноземні вчені як, Назаркевич М.А., Можаяєв О.О., Розорінов Г.М., Варфоломеев А., Антонюк О., Редмон Д., Сміт В., та інші.

Мета і завдання дослідження Метою дисертаційної роботи є удосконалення можливостей використання алгоритмів розпізнавання візуальної інформації для електронних систем безпеки приміщень або інших об'єктів мегаполісу, на основі запровадження інструментарію нейронних мереж. Основна увага має бути при цьому прикута на оцінці ефективності їх використання, особливо, при складних умовах видимості та освітлення, з забезпеченням при цьому, необхідної точності розпізнавання, швидкості виділення ознак та надійності роботи в цілому запропонованих систем безпеки в режимі реального часу.

Об'єктом дослідження є програмні алгоритми та методи розпізнавання візуальної інформації в електронних системах безпеки, які використовуються для ідентифікації об'єктів у реальному часі за умов складних умов оточуючого середовища фіксації.

Предметом дослідження є динамічне кольорове зображення або секвенція кадрів, отриманих з камери системи відеонагляду, за різних оточуючих умов фіксації.

Досягнення поставленої мети передбачає вирішення таких завдань:

1. Провести огляд і класифікацію існуючих алгоритмів розпізнавання візуальної інформації, включаючи традиційні методи та підходи на основі

нейронних мереж, із акцентом на формулювання їх переваг та обмежень для електронних систем безпеки.

2. Дослідити ефективність сучасних нейронних мереж, які реалізовані в моделях YOLOv7, для задач розпізнавання об'єктів у реальних умовах, та порівняти їх з традиційними методами обробки візуальної інформації.

3. Провести експериментальну оцінку ефективності застосування алгоритмів розпізнавання спеціальних об'єктів у складних умовах, таких як низька видимість, шум, часткове перекриття об'єктів та прозорість. Розробити нові підходи до їх покращення з метою підвищення точності та швидкості розпізнавання спеціальних об'єктів залежно від їх дальності розташування до камери системи відеонагляду.

Розглянуті в роботі підходи спрямовані на оцінку ефективності існуючих алгоритмів та їх вдосконалення, що дозволить розширити їх можливості у вирішенні актуальних завдань електронних систем безпеки.

Методи дослідження. Для досягнення поставленої мети використано методи комп'ютерного та об'єктно-орієнтованого моделювання, а також методи експериментального оцінювання можливостей удосконаленого алгоритму виявлення ознак об'єктів на зображенні при різних умовах фіксації.

Наукова новизна отриманих результатів. У дисертації представлено наступні наукові результати:

1. Вперше розроблено модифіковану систему розпізнавання зображень, яка дозволяє ідентифікувати об'єкти холодної зброї в інфрачервоному режимі та в умовах низької видимості.

2. Удосконалено роботу нейронної мережі згорткового типу, яка дозволяє при обробленні зображень зменшувати адитивний яскравісний шум зображення у випадку, коли сам рівень шуму може змінюватись в при аналізі секвенції зображень.

3. Удосконалено оптичну систему розпізнавання тексту, яка характеризується стійкістю до заважаючих фонових елементів у формі кривих Без'є.

4. Удосконалено архітектуру моделі YOLO V7 за рахунок додавання капсульного шару, внаслідок чого, отримано покращені результати виявлення об'єктів у складних сценах із частковим перекриттям або зміною форми останніх, а також в умовах зменшеної видимості об'єктів зображення за наявності та відсутності освітлення.

Особистий внесок здобувача.

Усі результати, наведені у дисертаційній роботі і винесені на захист, отримано за активної участі здобувача та опубліковано у спеціалізованих фахових виданнях.

В статті Переверзєв О.А., Трапезон К.О. Дослідження програмних алгоритмів відстеження руху об'єктів в електронних системах безпеки // Вчені записки Таврійського національного університету імені В.І.Вернадського. Серія: Технічні науки. Том 33(72), №6, 2022. С.288-293. DOI: <https://doi.org/10.32782/2663-5941/2022.6/47> опублікованій у співавторстві, здобувач провів порівняння двох алгоритмів відстеження об'єктів.

В статті Переверзєв О., Трапезон К. Особливості використання нейронних мереж у задачах зменшення шуму цифрових зображень // “Вісник КрНУ імені Михайла Остроградського”. – 2023. - №3 (140). – С. 44-51. DOI: <https://doi.org/10.32782/1995-0519.2023.3.6> (ISSN 1995–0519 print; ISSN 2072-8263 online), опублікованій у співавторстві, здобувач розробив та перевіряв лістинг програмного коду нейронної мережі.

В статті Переверзєв О., Трапезон К. Програмні підходи використання вейвлет-перетворення в задачах розпізнавання тексту із зображень // “Вісник КрНУ імені Михайла Остроградського”. – 2024. - №1 (144). – С. 266-272. DOI: <https://doi.org/10.32782/1995-0519.2024.1.35> (ISSN 1995–0519 print; ISSN 2072-8263 online), опублікованій у співавторстві, здобувач провів перевірку розробленого алгоритму на двох зображень, які отримано різними способами.

В статті Переверзєв О., Трапезон К. Принципи та підходи використання алгоритму YOLO v7 в системах виявлення об'єктів // “Вісник КрНУ імені Михайла Остроградського”. – 2024. - №4 (147). – С. 104-112. DOI:

<https://doi.org/10.32782/1995-0519.2024.4.13> (ISSN 1995–0519 print; ISSN 2072-8263 online), опублікованій у співавторстві, здобувач розробив програмні рішення з удосконалення архітектури нейронної мережі, яка підключена до моделі YOLO v7 та її модифікацій.

Практичне значення одержаних результатів.

1. Визначені рекомендації щодо оцінки моделей на основі YOLOv7 дозволяють удосконалити їхню роботу в умовах низького освітлення, зменшеної чіткості об'єктів та часткового перекриття, що є актуальним для систем, які працюють у складних умовах.

2. Отримані результати можуть бути інтегровані у прикладні рішення для автоматизованого виявлення зброї на інфрачервоних та оптичних зображеннях, що сприятиме у подальшому підвищенню рівня безпеки в громадських та стратегічно важливих об'єктах.

3. Запропоновані підходи до адаптації алгоритмів розпізнавання об'єктів можуть бути використані у військових системах, транспортних технологіях, інтелектуальних системах управління дорожнім рухом та інших галузях, де важливе своєчасне і точне розпізнавання об'єктів.

4. Практичні результати можуть знайти своє застосування у розробці програмного забезпечення для аналізу відеопотоку, орієнтованого на роботу з інфрачервоними камерами, що забезпечить можливість ефективного моніторингу в умовах поганої видимості.

Зв'язок роботи з науковими програмами, планами, темами.

Розглянуті у роботі нові практичні результати досліджень можуть бути використані при розробці удосконалених прикладних систем безпеки з застосування технологій Інтернету речей, а також у освітньому процесі при підготовці навчальних планів підготовки магістрів за освітньо-професійною програмою “Електронні системи мультимедіа та засоби Інтернету речей”.

Апробація результатів дисертації. Основні положення та результати дисертаційного дослідження доповідались на 1 міжнародній науково-практичній конференції:

Переверзев О., Трапезон К. Аналіз алгоритмів моніторингу об'єктів у системах безпеки // Матеріали IX Міжнародної науково-практичної конференції “Perspectives of Contemporary Science: Theory and Practice” (14–16 жовтня 2024 р., Львів, Україна). Львів, 2024.

Публікації. За результатами досліджень опубліковано 4 наукових публікацій (з них 3 статті у наукових фахових виданнях України за спеціальністю 171 Електроніка), 1 доповідь у збірниках матеріалів конференцій.

Структура та обсяг дисертаційної роботи. Дисертація складається зі вступу, чотирьох розділів, висновків, списку використаних джерел. Загальний обсяг роботи становить 156 сторінок, включаючи 39 рисунків, 16 таблиць і 108 посилань на джерела.

Список використаних джерел включає 108 найменувань

1 ОГЛЯД АЛГОРИТМІВ РОЗПІЗНАВАННЯ ВІЗУАЛЬНОЇ ІНФОРМАЦІЇ ДЛЯ СИСТЕМ БЕЗПЕКИ

1.1 Основні принципи та класифікація алгоритмів виявлення і відстеження об'єктів

Аналіз візуальної інформації при функціонуванні електронних систем безпеки, нерозривно пов'язаний з процедурами аналізу кольорового зображення, яке фіксується камерою цієї системи. Для реалізації цього аналізу відомо багато різних традиційних підходів та алгоритмів. Наприклад, можна відмітити процедури морфологічної обробки зображення, його сегментацію, фільтрацію, тощо. Разом з тим, ці підходи хоч і є простими і не вимагають значних обчислювальних ресурсів, проте мають ряд обмежень, які проявляються, коли саме зображення, з камери не є “ідеальним”. Наприклад, воно має яскравіший шум на поверхні, об'єкти на поверхні зображення перетинаються, або необхідно провести розпізнавання текстової інформації, проте оточуючий фон є складним з насиченими сторонніми зайвими елементами. Для успішного вирішення задач за цих умов, пропонується використовувати різні підходи, які пов'язані з удосконаленими алгоритмами відстеження об'єктів, алгоритмами з використанням вейвлет-перетворення або алгоритмами з залученням нейронних мереж згорткового типу.

З розвитком технологій машинного навчання алгоритми виявлення та відстеження об'єктів стали ключовими компонентами сучасних систем безпеки, транспортних технологій, військових систем і робототехніки. Їх основна мета полягає у швидкій та точній ідентифікації об'єктів, навіть у складних умовах, таких як зміна освітлення, фоновий шум чи часткове перекриття об'єктів на площині зображення.

Найбільш популярні сьогодні алгоритми виявлення об'єктів на зображенні базуються на двох основних підходах. Перший підхід передбачає аналіз вхідних даних, зокрема обробку окремих пікселів, виділення ключових точок

або класифікацію сукупностей пікселів, що відповідають за певний об'єкт. Наприклад, метод Optical Flow дозволяє визначати швидкість і напрямок руху об'єкта, базуючись на аналізі зміщення інтенсивності пікселів між кадрами [2]. Інший підхід зосереджений на динамічному оновленні моделей, які здатні адаптуватися до змін середовища, таких як освітлення, рух чи часткове перекриття об'єктів [3].

Класифікація алгоритмів за принципами їхньої роботи включає піксельні методи, які аналізують кожен піксель окремо, і об'єктні методи, що працюють із характеристиками об'єктів, такими як контури, текстури чи кольори. До прикладів піксельних методів належить Optical Flow, що особливо корисний для динамічних сцен [4]. Натомість об'єктні методи, як-от CAMShift, використовують кольорову модель HSV для відстеження об'єктів у статичних сценах [5].

Традиційні на сьогодні алгоритми, такі як CAMShift та Optical Flow, залишаються популярними завдяки своїй відносній простоті та ефективності в умовах обмежених ресурсів. Алгоритм CAMShift базується на аналізі кольорової інформації, тобто створює маску кольору та визначає центр ваги об'єкта, що дозволяє адаптувати область пошуку під час його руху. Основними перевагами є простота та швидкість, однак цей метод чутливий до змін освітлення та складного фону [6]. Алгоритм Optical Flow, який аналізує зміни інтенсивності пікселів між кадрами, і широко застосовується для задач із динамічним рухом, але вимагає значних обчислювальних ресурсів і є чутливим до шуму [4].

Розвиток глибинного навчання та згорткових нейронних мереж значно розширив можливості алгоритмів відстеження об'єктів. Одним із найуспішніших на сьогодні методів вважається YOLO (You Only Look Once), який забезпечує одночасне виявлення кількох об'єктів у реальному часі. Його ключовими перевагами є висока швидкість і точність, однак ефективність може знижуватися у випадках часткового перекриття об'єктів або слабого освітлення [7].

Серед ключових викликів, які стоять перед алгоритмами виявлення та відстеження об'єктів, можна виділити оклюзію, шум і складний фон. Оклюзія, тобто часткове перекриття об'єкта іншими елементами сцени, значно ускладнює його ідентифікацію. Наприклад, традиційні методи, такі як Optical Flow, часто втрачають слід об'єкта в таких умовах [8]. Шум, спричинений низькою якістю зображення або технічними перешкодами, може викликати помилкові спрацювання, але ця проблема частково вирішується за допомогою фільтрів, таких як гаусівський чи медіанний [9]. Складний фон із подібними текстурами або кольорами може вводити алгоритми в оману, особливо такі методи, як CAMShift, що базуються на аналізі кольорової моделі [10].

Розвиток комп'ютерного зору спрямований на інтеграцію традиційних алгоритмів із глибинним навчанням, що дозволяє поєднати переваги швидкості та точності нейронних мереж із адаптивністю класичних підходів. У майбутньому це сприятиме створенню більш ефективних і універсальних систем для розв'язання завдань у реальному часі [11].

1.2 Традиційні методи: алгоритми CAMShift, Optical Flow та їх характеристика

Традиційні методи відстеження рухомих об'єктів є основою для багатьох сучасних систем безпеки та відеоспостереження. До таких методів належать алгоритми CAMShift і Optical Flow, які широко використовуються завдяки своїй ефективності та простоті реалізації. Вони дозволяють визначати положення об'єкта та його переміщення між послідовними кадрами, що особливо важливо для моніторингу в реальному часі.

CAMShift (безперервний адаптивний зсув середнього) є модифікацією алгоритму Mean Shift і базується на аналізі кольорових характеристик об'єкта у з використанням певної кольорової моделі HSV. Ця модель забезпечує стабільність кольорових характеристик навіть за умов змін освітлення. CAMShift спочатку формує гістограму кольорового розподілу у вибраній

області, потім визначає умовний центр ваги цієї області, і, якщо центр зміщується, автоматично коригує межі пошуку об'єкта. Така адаптивність дозволяє відстежувати об'єкти, які змінюють розмір і форму під час руху [12], [13].

Рівняння для визначення центру ваги в CAMShift можна записати на основі формули (1.1).

$$M = \sum_{i=1}^n w_i x_i \quad (1.1)$$

де M — зсув, w_i — вага кольорового розподілу для i -го пікселя, а x_i — координата пікселя. Ця вага базується на значеннях гістограми кольору.

Алгоритм CAMShift добре підходить для задач із відстеження об'єктів із постійними кольоровими характеристиками та на фоні із низькими перешкодами. Проте цей метод має свої обмеження: він чутливий до схожості кольорів об'єкта і фону, що може призвести до втрати сліду у складних сценах [14].

На відміну від CAMShift, алгоритм Optical Flow базується на аналізі змін інтенсивності пікселів між послідовними кадрами. Основне припущення методу полягає в тому, що інтенсивність пікселів залишається сталою під час їхнього переміщення у просторі. Це дозволяє визначити напрямок і швидкість руху об'єкта, використовуючи рівняння оптичного потоку:

$$I_x V_x + I_y V_y = -I_t \quad (1.2)$$

де I_x і I_y — похідні яскравості по координатах X та Y , відповідно; I_t — похідна по часу; V_x та V_y — компоненти вектора швидкості руху [2].

Оптичний потік може бути реалізований за допомогою методу Lucas-Kanade, який аналізує зміни у фіксованих вікнах зображення. Цей підхід забезпечує високу точність відстеження, проте він вимагає значних обчислювальних ресурсів і може втратити слід об'єкта у складних сценах із великими фоновими перешкодами [3]. Основні характеристики алгоритмів CAMShift і Optical Flow зведені у Таблиці 1.1.

Таблиця 1.1 – Порівняльна таблиця характеристик CAMShift та Optical Flow

Характеристика	CAMShift	Optical Flow
Тип даних	Використовує кольорову інформацію	Працює на основі інтенсивності пікселів
Залежність від кольору	Висока (кольорова схема HSV)	Низька (аналізує інтенсивність, а не колір)
Стійкість до змін освітлення	Середня (залежить від стабільності кольору)	Висока (нечутливий до змін у кольоровій схемі)
Ресурси	Економічний з точки зору обчислень	Вимагає високих обчислювальних ресурсів
Середовище використання	Найкраще підходить для статичних сцен з малими фоновими перешкодами	Підходить для складних і динамічних сцен
Чутливість до шуму	Середня (залежить від контрастності кольору)	Висока (може втрачати об'єкт у складних сценах)
Переваги	Простий у реалізації, адаптивний до розмірів об'єкта	Точний для відстеження складних форм і напрямків руху об'єктів
Недоліки	Може втрачати об'єкт у разі змін кольору або освітлення	Чутливий до шуму та вимагає значних обчислень
Приклад застосування	Відстеження простих об'єктів у відеоспостереженні	Аналіз складних рухів у системах з високою вимогою до точності

Таким чином, алгоритми CAMShift і Optical Flow мають свої переваги та обмеження, що робить їх ефективними для різних типів задач. CAMShift підходить для простих об'єктів із стабільними кольоровими характеристиками,

тоді як Optical Flow має свої переваги в динамічних умовах. Вибір між ними залежить від вимог до точності, швидкості обчислень і ресурсів системи [10].

Подальший розвиток цих алгоритмів та їх інтеграція з методами глибокого навчання дозволяють створювати системи, які ефективно працюють у реальному часі навіть у складних умовах [15].

1.3 Підходи розпізнавання тексту на зображеннях

Розпізнавання тексту на зображеннях є ключовою задачею для багатьох сучасних систем, таких як, автоматизований аналіз документів, відеоспостереження та інтелектуальні транспортні системи. Одними з найпоширеніших алгоритмів для таких задач є використання вейвлет-перетворення та оптичного розпізнавання символів (OCR).

Вейвлет-перетворення є потужним інструментом для оброблення та редагування зображень, оскільки воно дозволяє аналізувати їх на різних масштабах і виділяти, наприклад дрібні деталі, такі як контури тексту. У задачах розпізнавання тексту це перетворення допомагає виділяти текстові елементи серед складного фону за рахунок розкладання зображення на частотні компоненти. Дискретне вейвлет-перетворення (DWT) дозволяє виділити високочастотні компоненти, які відповідають за дрібні деталі, такі як контури тексту. Завдяки цьому вейвлет-перетворення ефективно працює на зображеннях із неоднорідним фоном, мінімізуючи його вплив на видимість тексту.

Процес DWT описується наступною формулою [16]:

$$DWT(x) = \sum_{n=-\infty}^{\infty} x[n] \psi\left(\frac{n-k}{2^j}\right)$$

де $x[n]$ — значення зображення, ψ — функція вейвлета, j — рівень масштабування, а k — зсув. Це дозволяє виділити контури тексту та мінімізувати вплив фону [16], [17].

Вейвлет-перетворення може використовуватись і для зображень із складним фоном, де стандартні методи виділення тексту можуть бути недостатньо ефективними. Завдяки цьому перетворенні текстові елементи виділяються більш чітко, навіть якщо фон містить багато деталей [18], [19].

Оптичне розпізнавання символів (OCR) перетворює текст із зображень у цифровий формат. Алгоритм OCR складається з кількох етапів:

1. Попередня обробка зображення: покращення контрастності, зменшення шуму та бінаризація. Одним із найпоширеніших методів є метод Отсу, який визначає оптимальне порогове значення для поділу тексту та фону:

$$T = \arg \min_t (\sigma_w^2(t))$$

де T — порогове значення, що мінімізує дисперсію між класами, а $\sigma_w^2(t)$ — дисперсія всередині класів для порогу t [20], [21].

2. Сегментація та розпізнавання символів: зображення розбивається на окремі символи, які порівнюються з шаблонами або обробляються за допомогою моделей машинного навчання [22], [23].

При оптичному розпізнаванні тексту використовуються різні динамічні бібліотеки в сфері довжини об'єктно-орієнтованого програмування, наприклад, бібліотека Tesseract.

Нижче наведено в таблиці 1.2 порівняння основних характеристик методів вейвлет-перетворення та OCR.

Таблиця 1.2 — Порівняльна таблиця характеристик вейвлет-перетворення та OCR

Характеристика	Вейвлет-перетворення	OCR
Основне призначення	Виділення тексту на зображенні з урахуванням фону	Конвертація тексту в цифровий формат
Тип вхідних даних	Працює з текстом, що має контури та виділяється з фону	Підходить для друкованого тексту, може працювати з рукописом
Чутливість до фону	Чутливий до фону, що може вплинути на якість виділення	Не залежить від фону, якщо попередня обробка проведена якісно
Ресурси та швидкість	Вимагає обчислювальних ресурсів для обробки великого обсягу зображень	Швидкий та ефективний для практичних застосувань
Обмеження	Може давати хибні результати на складних фонах	Може вимагати додаткової обробки для рукописного тексту

Вейвлет-перетворення підходить для виділення текстових елементів на зображеннях із складним фоном, тоді як OCR використовується для перетворення тексту у цифровий формат для подальшого аналізу. У багатьох випадках ці методи використовуються разом: спочатку вейвлет-перетворення виділяє текст, а OCR розпізнає його, забезпечуючи високу точність навіть у складних умовах.

1.4 Проблеми оклюзії, шуму та фону в задачах розпізнавання об'єктів

В задачах розпізнавання об'єктів у режимі реальному часі важливо враховувати три основні фактори, які знижують точність систем технічного зору: оклюзію, шум і складний фон. Кожен із цих факторів по-різному впливає на процес розпізнавання, тому необхідно розробляти методи для їх мінімізації.

Оклюдія виникає, коли частина об'єкта, що відстежується, закривається іншим об'єктом або елементом фону. Це може призводити до тимчасової або повної втрати об'єкта. Оклюдія є особливою проблемою в багатолюдних місцях, де об'єкти часто перекривають один одного, або у складних сценах із численними перешкодами.

Для розв'язання цієї проблеми використовують алгоритми на основі нейронних мереж, зокрема згорткових нейронних мереж (CNN), які здатні визначати об'єкти навіть за часткової видимості [24], [8]. Також популярними є методи прогнозування положення об'єкта на основі попередніх кадрів із застосуванням розпізнавання особливих точок, що знижує залежність від постійної видимості об'єкта.

Шум є ще одним фактором, який негативно впливає на точність розпізнавання. Він виникає через низьку якість зображення, особливості освітлення або технічні характеристики камери. Наявність шуму може ускладнювати виявлення контурів об'єктів або спричиняти помилкові спрацьовування.

Для зменшення шуму часто використовуються фільтри, такі як медіанний або гауссівський, які згладжують зображення та знижують кількість випадкових перешкод. Наприклад, гауссівський фільтр описується рівнянням [9]:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

де σ — стандартне відхилення, що визначає ступінь згладжування. Фільтрація допомагає приглушити шум і виділити об'єкт, але може знизити чіткість контурів [25], [26].

Додатково використовують нейронні мережі, що застосовують за своєю архітектурою згорткові шари для зменшення білого шуму. Глибокі нейронні мережі використовують стратегію залишкового навчання, яка дозволяє підвищити якість зображення шляхом видалення шуму без втрати важливих деталей [9], [27].

Складний фон також може негативно впливати на точність систем розпізнавання об'єктів, особливо якщо фон містить схожі елементи або текстури. Наприклад, у міському середовищі фон може включати об'єкти зі схожими формами чи кольорами, що ускладнює розпізнавання.

Для зменшення впливу складного фону застосовують методи сегментації та маскування, які дозволяють виділити об'єкт і зменшити вплив фонових елементів. Аналіз руху також допомагає відокремити об'єкт за його переміщенням, зменшуючи вплив статичного фону. У цьому контексті ефективними є нейронні мережі, здатні виділяти особливі ознаки об'єктів [28], [29].

Зведення основних характеристик оклюзії, шуму та фону, а також методів мінімізації, подано в таблиці 1.3.

Таблиця 1.3 — Таблиця факторів впливу на точність та методи мінімізації

Фактор	Вплив на точність	Методи мінімізації	Приклади алгоритмів
Оклюдія	Високий	Прогнозування руху, нейронні мережі	YOLO, SSD
Шум	Середній	Фільтрація (медіанна, гауссівська), CNN	Гауссівське згладжування, CNN
Фон	Високий	Сегментація, аналіз руху, виділення об'єкта	R-CNN, методи аналізу руху

Таблиця 1.3 наочно ілюструє, як різні фактори впливають на точність систем розпізнавання об'єктів, а також ефективні підходи для їх мінімізації. За допомогою сучасних алгоритмів, таких як згорткові нейронні мережі, можна значно підвищити стійкість систем до впливу оклюзії, шуму та фону, що робить їх придатними для роботи в складних умовах.

Проблеми оклюзії, шуму та фону є важливими аспектами, які необхідно враховувати при розробці алгоритмів розпізнавання об'єктів. Сучасні підходи, що включають нейронні мережі та методи фільтрації, значно підвищують стійкість алгоритмів до цих факторів, забезпечуючи високу якість розпізнавання навіть у складних умовах. Використання згорткових нейронних мереж є перспективним підходом для розпізнавання об'єктів, оскільки вони демонструють високу ефективність у задачах із шумними даними або на складному фоні [30], [31].

1.5 Порівняння нейронних мереж із традиційними методами обробки зображень

З розвитком технологій обробки зображень нейронні мережі, особливо згорткові нейронні мережі, стали на сьогодні основою для багатьох сучасних додатків комп'ютерного зору, таких як розпізнавання обличч, автономне

водіння автомобіля та системи безпеки. Водночас, традиційні методи обробки зображень залишаються актуальними для задач, що потребують швидкої обробки з мінімальними ресурсами. У цьому розділі порівнюються основні характеристики та переваги обох підходів.

Традиційні методи, про які йшла мова вище в роботі, базуються на алгоритмічних підходах щодо виділення ознак, таких як краї, текстури або контури зображення. Основні методи включають:

- **фільтри країв:** Використовуються для визначення контурів на зображенні, наприклад, алгоритми Canny та Sobel [32], [33]. Ефективні для простих задач виділення контурів у статичних умовах.
- **вейвлет-перетворення:** Дозволяє виділяти деталі зображення на різних масштабах і використовується для задач, як-от виділення текстових елементів на складному фоні [16].
- **оптичний потік:** Використовується для аналізу руху на послідовності кадрів, визначаючи напрямок і швидкість переміщення об'єктів [3].

Традиційні методи мають такі переваги:

- **низькі обчислювальні вимоги:** Вони зазвичай менш ресурсомісткі, що дозволяє виконувати обробку на пристроях з обмеженою потужністю.
- **простота реалізації:** Багато з цих методів легко реалізуються і швидко виконуються навіть на простих комп'ютерах.

Разом з тим, традиційні методи мають і певні обмеження:

- **чутливість до шуму:** Вони можуть бути чутливими до шуму і зміни освітлення, що знижує їхню ефективність у реальних умовах [13].
- **обмежена точність:** Традиційні методи часто не мають можливості глибокого аналізу ідентифікаційних ознак об'єктів, що знижує точність у складних задачах, таких як розпізнавання облич або детекція дрібних об'єктів [13].

Нейронні мережі, зокрема CNN, є більш складним і точним інструментом для обробки зображень. Вони дозволяють автоматично вивчати ознаки об'єктів із великих наборів даних і ефективно адаптуватися до різних умов.

- автоматичне виділення ознак: CNN самостійно вивчають релевантні ознаки на основі навчальних даних, що значно зменшує потребу в ручному налаштуванні [34].
- висока точність: Нейронні мережі демонструють високу точність навіть у складних умовах, як-от шум або зміни освітлення [9].
- стійкість до змін: CNN адаптуються до різних масштабів, поворотів і освітлення завдяки навчанням на великих наборах даних [9].

Однак нейронні мережі мають і обмеження:

- високі обчислювальні ресурси: Навчання нейронних мереж вимагає значних обчислювальних потужностей та часу, особливо для глибоких архітектур, таких як YOLO або ResNet.
- необхідність великих навчальних наборів даних: Для досягнення високої точності нейронні мережі потребують значної кількості даних для навчання, що може бути складним у разі обмеженого доступу до мічених даних.

Для кращого розуміння переваг та недоліків кожного підходу наведемо порівняння в Таблиці 1.4.

Таблиця 1.4 — Порівняння традиційних методів та нейронних мереж

Параметр	Традиційні методи	Нейронні мережі
Вимоги до обчислювальних ресурсів	Низькі	Високі
Точність	Середня	Висока
Автоматичне виділення ознак	Ні	Так

Продовження таблиці 1.4 — Порівняння традиційних методів та нейронних мереж

Параметр	Традиційні методи	Нейронні мережі
Стійкість до змін освітлення	Низька	Висока
Чутливість до шуму	Висока	Низька
Потреба у великих даних	Низька	Висока
Швидкість обробки	Висока (на окремих кадрах або зображеннях)	Низька, але оптимізована для роботи з відеопотоками

Традиційні методи залишаються актуальними для простих задач, де важливі швидкість та обмежені обчислювальні ресурси. Однак нейронні мережі демонструють значно більшу гнучкість і точність у складних умовах, таких як зміни освітлення, шум або необхідність розпізнавання дрібних деталей. Завдяки своїм перевагам CNN стали основним інструментом для вирішення задач комп'ютерного зору, що потребують високої продуктивності та адаптивності [34].

1.6 Модель YOLO для виявлення об'єктів на зображеннях

Модель YOLO (You Only Look Once) є однією з провідних архітектур для виявлення об'єктів на зображеннях у реальному часі. Головна ідея YOLO полягає в обробці зображення за один прохід через мережу, що забезпечує швидке та ефективне ідентифікування й локалізацію об'єктів. Завдяки цьому YOLO широко використовується в таких сферах, як відеоспостереження, автономне водіння, аналіз аерофотозйомки тощо [35], [36].

Модель YOLO базується на ключових елементах архітектури CNN, таких як згорткові шари, шари підвибірки та пов'язані шари, описаних у

попередньому розділі. Однак YOLO має низку унікальних особливостей, які роблять її придатною для швидкого та точного виявлення об'єктів.

Основні принципи архітектури YOLO наступний.

1. Поділ зображення на сітку: YOLO ділить вхідне зображення на сітку розміром $S \times S$. Кожна клітинка відповідає за передбачення об'єктів, які знаходяться в її межах. Кожна клітинка генерує кілька обмежувальних рамок, зокрема параметри рамки (координати центру x , y , ширина w та висота h), клас об'єкта та впевненість у наявності об'єкта [7]:

$$c = P(\text{об'єкт}) \cdot IOU_{\text{прогнозна рамка, істинна рамка}}$$

де IOU (Intersection Over Union) показує частку перетину обмежувальних рамок щодо їх об'єднання [37].

2. Прогноз обмежувальних рамок (Bounding Boxes): Для кожної клітинки модель генерує декілька обмежувальних рамок. Всього для кожного зображення отримується $S \times S \times B$ обмежувальних рамок. Для кожної рамки оцінюється впевненість у наявності об'єкта, що дозволяє моделі визначати точне положення та розміри об'єктів [38].

3. Класифікація об'єктів: YOLO одночасно виконує задачі класифікації та локалізації, визначаючи клас об'єкта й координати обмежувальної рамки за один прохід. Ймовірність класу об'єкта визначається як:

$$P(c_i | \text{об'єкт}),$$

де $P(c_i)$ — ймовірність належності об'єкта до класу c_i

Базовий алгоритм роботи моделі YOLO включає наступні етапи:

- Попередня обробка зображення: Зображення масштабується до фіксованого розміру, наприклад 416×416 пікселів, щоб забезпечити узгодженість вхідних даних.
- Прогноз обмежувальних рамок і класів: Мережа прогнозує обмежувальні рамки та класи об'єктів для кожної клітинки сітки, що дозволяє одночасно локалізувати та ідентифікувати об'єкти.

- Не-максимальне придушення (Non-Maximum Suppression, NMS): Для усунення перекриття обмежувальних рамок використовується NMS, що залишає лише ті рамки, які мають найвищу впевненість.

За роки розвитку алгоритм YOLO пройшов кілька удосконалень:

- YOLOv3: Додано багатомасштабне прогнозування, що покращило точність для малих об'єктів [7].

- YOLOv4: Введено CSPDarknet, що підвищило ефективність передачі градієнтів у глибоких мережах. Крім того, додано нові оптимізації, такі як Mish-активація, Mosaic Data Augmentation і DropBlock Regularization, що суттєво покращило як швидкість, так і точність моделі [39].

- YOLOv5: Основною особливістю YOLOv5 є простота впровадження та оптимізація для роботи на пристроях із різною продуктивністю. YOLOv5 отримала значну популярність завдяки зручності використання, інтегрованій підтримці PyTorch, ефективній тренувальній процедурі та можливості масштабування (від легких моделей для мобільних пристроїв до великих для серверів). Було покращено швидкість інференсу та впроваджено нові функції, такі як автоматичне масштабування навчальних гіперпараметрів і вдосконалений алгоритм Data Augmentation [40].

- YOLOv6: Орієнтована на індустріальне застосування, YOLOv6 поєднує легковагу архітектуру з вдосконаленими методами передобробки даних, що дозволяє працювати на пристроях із обмеженими ресурсами. Основними інноваціями є новий Head-блок і оптимізовані модулі обчислень, що роблять YOLOv6 надзвичайно продуктивною [41].

- YOLOv7: Додано сегментацію, покращену локалізацію та модуль CBAM (Convolutional Block Attention Module), що дозволяє моделі фокусуватися на важливих ділянках зображення. YOLOv7 покращила точність і швидкість роботи, особливо в задачах, що вимагають багатозадачності [36].

- YOLOv8: Ця версія базується на архітектурі YOLOv5 та включає численні покращення як в архітектурі, так і в зручності для розробників. YOLOv8 здобула популярність завдяки оптимальному поєднанню швидкості та

точності, забезпечуючи швидший інференс при збереженні високої продуктивності в реальному часі. Особливо актуальною вона стала для застосувань із низькою затримкою [42].

- YOLOv9: Ця версія впроваджує подальші покращення в швидкості, точності та обчислювальній ефективності порівняно з YOLOv8. В YOLOv9 використано новітні алгоритми та оптимізації, зокрема Generalized Efficient Layer Aggregation Network (GELAN) та Programmable Gradient Information (PGI), що значно підвищують здатність моделі до вилучення ознак та покращують потік градієнтів, забезпечуючи вищу точність та ефективність. Тестування на наборі даних Microsoft COCO показало, що YOLOv9 перевершує YOLOv8 за кількома показниками, демонструючи вищу середню точність (mAP) та швидший інференс [43].

Переваги YOLO:

- швидкість: YOLO обробляє зображення за один прохід, що робить його ефективним для застосувань у реальному часі, таких як відеоспостереження.
- гнучкість: Підтримка різних версій для різних задач – від детекції дрібних об'єктів до аналізу складних динамічних сцен [44].

Обмеження YOLO наступні:

- Чутливість до дрібних об'єктів: Ранні версії YOLO мали труднощі з точним визначенням дрібних об'єктів.
- Нестабільність на складних сценах: На насичених зображеннях модель може давати хибнопозитивні результати [45].

YOLO залишається на сьогодні однією з найефективніших архітектур для виявлення об'єктів завдяки здатності одночасно виконувати локалізацію та класифікацію.

1.7 Прикладні задачі застосування нейронних мереж

1.7.1. Можливості нейронних мереж у системах відеоспостереження та контролю дорожнього трафіку

Нейронні мережі, особливо згорткові нейронні мережі (Convolutional Neural Networks, CNN), є ключовими інструментами в сучасних системах відеоспостереження та контролю трафіку. Завдяки своїй здатності автоматично виділяти суттєві ознаки та аналізувати дані в реальному часі, вони забезпечують високий рівень точності та ефективності. У цьому розділі розглядаються можливості нейронних мереж для виявлення, відстеження та класифікації об'єктів, а також їх застосування у відеоспостереженні та управлінні дорожнім рухом.

У задачах відеоспостереження та контролю дорожнього трафіку однією з ключових задач є виявлення об'єктів (наприклад, автомобілів, пішоходів, велосипедистів). Нейронні мережі, такі як YOLO (You Only Look Once) і Faster R-CNN, досягли значних успіхів у цій галузі завдяки одночасній локалізації та класифікації об'єктів [46].

- **YOLO:** Швидкість та ефективність YOLO дозволяють використовувати його в реальному часі для моніторингу безпеки в публічних місцях або виявлення порушень правил дорожнього руху. Наприклад, модель ділить зображення на сітку й обробляє його за один прохід, що забезпечує швидке визначення об'єктів [46].

- **Faster R-CNN:** Використовує регіональні пропозиційні мережі (RPN), що дозволяють забезпечити високу точність виявлення навіть за складних умов [11].

Для систем контролю дорожнього трафіку важливо не лише виявити об'єкт, але й відстежити його рух на послідовності кадрів. Методи, як-от Deep SORT (Simple Online and Realtime Tracking), у поєднанні з CNN дозволяють

ефективно відстежувати об'єкти в динамічних сценах, наприклад, на складних перехрестях або в умовах інтенсивного руху [47].

- **Відстеження руху:** Відстеження дозволяє аналізувати поведінку об'єктів, таких як транспортні засоби або пішоходи. Наприклад: Виявлення порушення правил (наприклад, заїзд у заборонену зону), прогнозування аварійних ситуацій на основі аналізу траєкторій, визначення швидкості руху або перетину розмітки.

Класифікація об'єктів є важливим завданням для систем контролю трафіку. CNN, такі як ResNet і Inception, забезпечують високу точність завдяки своїй здатності виділяти ключові ознаки зображень [48].

- **Розпізнавання типів транспортних засобів:** CNN дозволяють ідентифікувати транспортні засоби за типами (наприклад, автомобілі, вантажівки, мотоцикли), що є корисним для аналізу руху.

- **Розпізнавання номерних знаків:** Для точного розпізнавання номерних знаків використовуються комбінації CNN і рекурентних нейронних мереж (RNN), що забезпечує точність навіть за поганих умов освітлення чи високої швидкості руху транспортного засобу [49].

Сучасні системи контролю трафіку використовують нейронні мережі для аналізу транспортних потоків у реальному часі. Наприклад:

- **Оцінка інтенсивності руху:** CNN у поєднанні з методами рекурентного навчання, такими як LSTM, використовуються для прогнозування заторів та аналізу потоків даних із камер [50].

- **Моніторинг ситуації на дорогах:** Системи здатні виявляти місця з високим ризиком аварій та надавати рекомендації для їх уникнення.

Переваги нейронних мереж у відеоспостереженні та контролі трафіку

- **Висока точність:** Нейронні мережі забезпечують точне розпізнавання різних типів об'єктів навіть за несприятливих умов, таких як шум або низька якість зображень.

- Швидкість обробки: Архітектури, як-от YOLO, забезпечують обробку зображень у реальному часі, що критично важливо для систем відеоспостереження [46].
- Масштабованість: Нейронні мережі адаптуються до обробки великої кількості відеопотоків та різних умов.
- Адаптивність: Моделі здатні навчатися на нових даних, що забезпечує їх ефективність у змінних умовах, таких як різні погодні умови або час доби [11].

Незважаючи на значні переваги, використання нейронних мереж у відеоспостереженні та контролі трафіку супроводжується певними викликами:

- Високі обчислювальні ресурси: Навчання та інференс потребують потужного апаратного забезпечення, що може бути викликом для великих систем [51].
- Потреба у великих наборах даних: Ефективність залежить від якості та кількості даних для навчання.
- Чутливість до змін якості відео: Зниження роздільної здатності чи наявність великого шуму можуть зменшити точність.

Нейронні мережі забезпечують високу точність, адаптивність і швидкість, що робить їх незамінними інструментами для сучасних систем відеоспостереження та управління дорожнім рухом. Вони дозволяють вирішувати широкий спектр задач: від виявлення об'єктів і класифікації до аналізу поведінки та прогнозування аварійних ситуацій. Завдяки своїй гнучкості нейронні мережі є основою для майбутнього розвитку інтелектуальних транспортних систем.

1.7.2 Особливості нейронних мереж у військових системах

Нейронні мережі стали невід'ємною частиною сучасних військових систем завдяки їхній здатності обробляти великі обсяги даних, забезпечувати високу точність у критичних задачах і автоматизувати складні процеси. У

військовій сфері вони знаходять застосування у розпізнаванні об'єктів, аналізі відеопотоків, прогнозуванні та підтримці прийняття рішень.

Однією з ключових задач нейронних мереж у військових системах є виявлення та класифікація об'єктів, таких як військова техніка, транспортні засоби, літальні апарати та потенційні загрози. Наприклад, моделі YOLO демонструють високу швидкість і точність у реальному часі, що є критичним для ефективного реагування в умовах бойових дій [46]. Faster R-CNN забезпечує високу точність навіть у складних умовах, таких як низька видимість чи наявність шуму [11].

Окрім цього, нейронні мережі здатні оцінювати стан техніки, виявляючи пошкодження чи знос, що дозволяє прогнозувати потреби в обслуговуванні. CNN також ефективні для класифікації об'єктів у динамічних сценах, що дозволяє відстежувати рухомі об'єкти навіть на великих відстанях.

Для військових систем, таких як БПЛА чи радарні системи, важливим завданням є не лише виявлення, а й відстеження об'єктів у динамічних умовах. Алгоритми Deep SORT дозволяють відстежувати рухомі об'єкти в реальному часі, що критично для моніторингу периметра чи стеження за ворожою технікою [47].

Нейронні мережі також використовуються для прогнозування траєкторій об'єктів, що допомагає виявляти потенційні загрози, такі як наближення літального апарату до забороненої зони або переміщення ворожих підрозділів [52].

Нейронні мережі широко використовуються для систем ідентифікації осіб та військової техніки:

- Розпізнавання обличь і біометричних даних: Сучасні моделі CNN, такі як FaceNet, можуть точно ідентифікувати осіб навіть за низької якості зображень, що корисно для контролю доступу до військових об'єктів або ідентифікації ворожих агентів [53].

- Класифікація техніки: Нейронні мережі здатні розпізнавати тип військової техніки (наприклад, танки або літаки), що допомагає ідентифікувати цілі на полі бою.

CNN застосовуються для обробки радіолокаційних даних. Наприклад, розпізнавання спектрограм для класифікації об'єктів, таких як літальні апарати чи сигнали радіолокаційних станцій, є одним із ключових напрямів використання [34]. Цей підхід дозволяє автоматизувати процеси виявлення сигналів і раннього попередження у військових операціях.

Рекурентні нейронні мережі (RNN), такі як LSTM, використовуються для прогнозування подій на основі історичних даних. Це дозволяє аналізувати переміщення ворога, прогнозувати загрози та оптимізувати маршрути військових операцій (таблиця 1.5) [54].

Таблиця 1.5 — Переваги та обмеження використання нейронних мереж у військових системах

Переваги	Обмеження
Висока точність і надійність, навіть у складних умовах.	Потреба у високих обчислювальних ресурсах.
Швидкість обробки, що дозволяє працювати в реальному часі.	Залежність від великих навчальних наборів даних.
Можливість роботи з низькоякісними даними.	Відсутність прозорості: “чорний ящик” складно пояснити в критичних умовах [55].

Нейронні мережі є невід'ємною частиною сучасних військових систем завдяки своїй здатності аналізувати великі обсяги даних, автоматизувати складні задачі та підвищувати рівень точності в критичних умовах. Попри виклики, такі як високі вимоги до апаратного забезпечення чи залежність від навчальних даних, їх використання стає ключовим фактором у розвитку військових технологій.

В рамках практичної частини дослідження на основі однієї з модифікацій моделей YOLO з запровадженими змін в її функціонуванні, перевіримо як ці інструменти дозволяють вирішити одну з нових задач, які пов'язані з роботою камер відеонагляду. Насамперед, як модель дозволяє ідентифікувати об'єкти у вигляді холодної зброї. Іншими словами, за рахунок створення нової системи (класу) класифікації об'єктів визначимо, як їх ідентифікація проходить і з якою точністю, коли за периметром наявне погане освітлення приміщення або коли штучно зловмисником створено умови до погіршення цього освітлення (розпилювання газів чи хімічних сумішів).

За результатами проведеного літературного огляду можна відзначити, широкі перспективи застосування нейронних мереж в області оброблення динамічних кольорових зображень. Це напевно, пояснюється тим, що цей інструментарій має свої переваги перед традиційними способами оброблення зображень. Так, окрім роботи в режимі реального часу, нейронні мережі можуть покращувати свої результати роботи, через те, що вони проходять обов'язково етап навчання на широкій вибірці навчальних даних. Разом з тим, відмічено, що на ефективність роботи систем з підтримкою нейронних мереж при обробці та аналізі зображень можуть впливати оклюзія, шум зображення та складний фон.

Висновки до розділу 1.

1. Принципи побудови нейронних мереж згорткового типу: згорткові нейронні мережі (CNN) є основою сучасних систем обробки зображень, що дозволяють автоматично виділяти ключові ознаки, такі як контури, текстури та структури об'єктів. Їх архітектура забезпечує високу точність і адаптивність до складних задач, зокрема завдяки використанню згорткових шарів, шарів підвибірки, залишкових блоків та модулів уваги.

2. Архітектура та алгоритми YOLO для виявлення об'єктів на зображеннях: модель YOLO (You Only Look Once) є ефективною архітектурою для задач одночасного виявлення й класифікації об'єктів у реальному часі. Її

ключові переваги — швидкість, можливість багатозадачної роботи та висока точність. Розвиток версій YOLO, зокрема YOLOv7, забезпечив додаткове покращення локалізації об'єктів, зменшення хибнопозитивних спрацювань і підвищення ефективності в складних умовах.

3. Порівняння нейронних мереж із традиційними методами обробки зображень: Нейронні мережі значно перевершують традиційні алгоритми, такі як CAMShift чи Optical Flow, завдяки своїй здатності працювати з великими обсягами даних, автоматично виділяти ознаки та демонструвати стійкість до змін освітлення, шуму й складного фону. Проте вони потребують більше обчислювальних ресурсів і навчальних даних.

4. Можливості нейронних мереж у системах відеоспостереження та контролю трафіку: нейронні мережі, такі як YOLO та Faster R-CNN, забезпечують високу точність і швидкість аналізу відеопотоків у реальному часі, що є критичним для задач відеоспостереження, контролю дорожнього руху та аналізу поведінки об'єктів. Їх застосування дозволяє автоматизувати виявлення порушень, аналіз трафіку та ідентифікацію об'єктів.

5. Особливості нейронних мереж у військових системах: застосування нейронних мереж у військових системах демонструє їхню ефективність у задачах розпізнавання об'єктів, таких як техніка або зброя, навіть за умов низької видимості чи сильного шуму. Завдяки стійкості до змін середовища та здатності працювати з різними типами даних (оптичні, інфрачервоні зображення) нейронні мережі є перспективним інструментом для підвищення функціональності військових систем.

2 ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА ДОСЛІДЖЕННЯ

В рамках першої практичної частини дослідження роботи і для того щоб оцінити можливості використання нейронних мереж та моделі YOLO на її основі щодо динамічних зображень для систем безпеки були сформульовані наступні задачі:

- перевірити як алгоритми CAMShift та Optical Flow працюють у реальних умовах для задач відстеження реальних об'єктів;
- для розробленої архітектури глибокої нейронної мережі перевірити спроможність видалення яскравісного шуму з кольорових зображень, за умови що сама мережа має пройти крізь процедуру навчання;
- перевірити розроблену схему алгоритму виділення та розпізнавання тексту на основі алгоритму OCR та застосування вейвлет-перетворення для випадку, коли сам текст нанесено на неоднорідний фон;
- дослідити можливості функціонування моделі YOLO v7 для зображень, які містять об'єкти з різними розмірами та характеризуються складним фоном.

2.1 Оцінка якості відстеження об'єктів на основі алгоритмів CAMShift та Optical Flow

На сьогоднішній день, зважаючи на стрімкий розвиток інформаційних технологій, використання комп'ютерних систем для створення зображень за допомогою алгоритмів виявлення рухомих об'єктів у реальному часі вже не є новинкою [56], [57]. Такі алгоритми активно застосовуються у різних галузях науки та техніки, включаючи інтерактивні відео, комп'ютерні ігри, військові системи трекінгу та роботизовані платформи. Однак, коли йдеться про відстеження людини або групи осіб у натовпі, завдання відеоспостереження значно ускладнюється через високі вимоги до швидкості обробки та точності розпізнавання об'єктів, що базуються на методах ідентифікації унікальних характеристик [6], [58].

Відеоспостереження за об'єктами розглядається як процес визначення їх точного місцезнаходження протягом певного часу шляхом аналізу відео або зображень із камери у реальному часі. З розвитком технологій комп'ютерного зору було запропоновано різні методи розв'язання проблеми відстеження людей. Наприклад, у роботах [6], [58] досліджується використання лазерного далекоміра для аналізу рухів людини та її анатомічних особливостей, таких як контури ніг. Проте цей підхід має обмеження, особливо при збільшенні відстані між об'єктом і датчиком. Також існують системи, що використовують кілька камер, але вони ефективні лише в обмеженому просторі [10], [15]

Ще одним методом є застосування ручних камер, як у дослідженні [15], для створення стробоскопічних зображень рухомих об'єктів. Проте цей підхід має свої недоліки, зокрема пов'язані зі швидкістю обчислень та точністю в складних умовах. Серед головних проблем, з якими стикаються алгоритми відстеження, можна виділити проблему оклюзії, коли фон або передній план перекриває об'єкт, що ускладнює його коректне відстеження [10], [15]. Це може спричиняти ситуації, коли один і той самий об'єкт помилково визначається як новий.

Ще одна важлива проблема – вплив фону. Чим складніший фон, тим важче виокремити характерні ознаки об'єкта, що може призводити до помилок у процесі аналізу [10], [15]. Надмірна кількість деталей у фоні може створювати шум, що негативно впливає на точність і швидкість роботи системи відеоспостереження.

Алгоритм відстеження не є алгоритмом окремого завдання, як це можна описати при класифікації зображень і виявленні об'єктів. Це насправді багатофункціональний алгоритм, який вирішує задачі виявлення об'єктів, їх локалізацію, класифікацію, а також відстеження за ознаками. Такими чином, на основі розглянутих проблем, які виникають при відстеженні рухомих об'єктів, виникає потреба в описі та аналізі методу відстеження, який б за основними особливостями дозволив б нівелювати зазначені обмеження.

На жаль в статті [56] розглянений алгоритм CAMShift використано лише для одного класу об'єктів, а саме задля відстеження людей і відповідно до цього, авторами визначені коефіцієнти на основі зваженої орієнтації градієнта за умови наявної інформації про колір. При цьому варто відмітити, що умови за якими фіксуються контури людей не є складними (не розглянуто ситуацію, коли камера відеонагляду змінює своє фокусування з одного об'єкту на інший в режимі реального часу), оскільки на основі кольорової інформації легко відрізняються від інших об'єктів на зображенні. Також в статті не розглянуто питання, коли фон цільового об'єкту перетинається з іншими об'єктами і коли цей фон співпадає повністю за кольорами з оточуючим фоном площини зображення. Крім цього в роботі не розглянуто алгоритм Optical Flow.

В роботі [57] для розпізнавання об'єктів пропонується використовувати згорткові нейронні мережі, проте основний нахил авторами робиться на аналізі стереозображень, які фіксуються з камер окремого виробника електронного обладнання. Крім цього, в даній роботі не розглянуто алгоритми CAMShift та Optical Flow, а визначені підходи важко адаптувати до задач відстеження об'єктів для випадків, коли використання згорткової нейронної мережі є неможливим.

Робота [58] описує способи використання лазерних технологій для вирішення задач сегментації об'єктів, в даному випадку ніг людей на вулиці за різних практичних умов. Використано алгоритм AdaBoost з формуванням відповідного класифікатора ознак. Проте, з даної роботи зовсім не зрозуміло, як можна використати цей алгоритм для розв'язання задачі відстеження різних об'єктів в умовах складного оточуючого фону.

На відміну від зазначених статей, у цій роботі здійснено комплексний аналіз алгоритмів CAMShift та Optical Flow в контексті їх стабільності та точності у різних умовах освітлення, шуму та форми об'єкта. Проведене порівняння дозволяє визначити, який алгоритм краще підходить для конкретних сценаріїв, що робить отримані результати практично значущими для розробників електронних систем безпеки. Завдяки врахуванню впливу

шуму, кольору та освітлення на роботу алгоритмів, отримані результати дають змогу точніше оцінювати їх продуктивність та потенційні обмеження. Таким чином, ця робота пропонує не лише поглиблений теоретичний аналіз, але й практичні рекомендації щодо застосування алгоритмів трекінгу в умовах реальних систем безпеки.

Враховуючи наведені теоретичні відомості у п.1.2 роботи, проведемо перевірку ефективності двох найбільш поширених алгоритмів CAMShift та Optical Flow щодо відстеження рухомих об'єктів на поверхні обраних динамічних кадрів зображення.

В роботі [59] проведено експериментальну оцінку ефективності алгоритмів CAMShift і Optical Flow для задач відстеження рухомих об'єктів. Головна мета в проведеному дослідженні полягала у порівнянні точності, швидкості роботи та стійкості цих двох алгоритмів до змінних умов, зокрема при наявності складного оточуючого фону.

Для реалізації алгоритму CAMShift використано дві програмні бібліотеки з середовища інструментів PyCharm – NumPy та OpenCV. Для обраного зображення або секвенції зображень і в такому випадку обирається один з кадрів цієї секвенції – референсний, для якого встановлюється початок області пошуку (область зображення, в якій розташовано об'єкт, що відстежується). Точніше за командами, які наведено в лістингу роботи [59] визначено область пошуку трекінгу. Дуже важливо для роботи алгоритму, щоб розмір області пошуку хоча б приблизно відповідав розміру об'єкта, який відстежується в площині зображення. За алгоритмом передбачається використання однієї з кольорових моделей, насамперед обрано базову кольорову модель HSV, про яку йшла мова в п.2.1 роботи. Далі створюється “маска” пошуку за якою відбувається пошук пікселів, які належать об'єкту. Особливістю цього алгоритму є те, що потрібно встановити у налаштуваннях умови, за якими відслідковування об'єкту зупиняється. На рисунку 2.1 показано результат відпрацювання цього алгоритму для секвенції кадрів, два кадри (перший та

останній) з яких наведено на цьому рисунку. В якості об'єкту обрано паперовий стаканчик і основними параметрами контролю визначено такі:

- точність визначення об'єктів (метрика Intersection Over Union, IOU).
- швидкість обробки (кількість кадрів за секунду).
- стійкість до змін умов у кадрах секвенції (освітлення, динаміка, різкість).



а)

б)

Рисунок 2.1 — Відстеження об'єктів за алгоритмом CAMShift у статичних умовах [59]: а) – перший кадр секвенції; б) – останній кадр секвенції

За отриманими даними для алгоритму CAMShift можна відмітити, що хоча алгоритм забезпечує швидке та адаптивне відстеження об'єктів у реальному часі, проте він має і особливості, а саме чутливість до змін кольорових характеристик об'єкта та освітлення.

Для даної тестової секвенції отримано, що область відстеження в кінцевому кадрі є менш точною, оскільки різкість кадру зменшується. Особливо це проявляється за нижніми рамками області відстеження.

Алгоритм Optical Flow натомість, має трохи інший алгоритм відстеження, де використовується поняття “особливих” крапок зображення. Відмітимо, що алгоритм реалізований на основі методу Lucas-Kanade, який відстежує вектори руху між кадрами (вектори зміщень). Параметри контролю обрані такі самі як і для алгоритма CAMShift. Метод Лукаса-Канаде, описаний у формулі (1.2), через бібліотеку OpenCV передбачає, що спочатку необхідно визначити функцію, на основі якою відбувається запуск відслідковування з використанням оптичного потоку [59]. Крім цього, за алгоритмом проводиться ініціалізація

об'єкту дослідження, визначаються параметри оптичного детектора на основі вбудованого алгоритму Shi-Tomasi і так само, як і для першого алгоритму, визначаються параметри кольорової моделі та задається маска області. На рисунку 2.2 наведено результат відстеження рухомого об'єкту автомобіля при фіксації з камери.

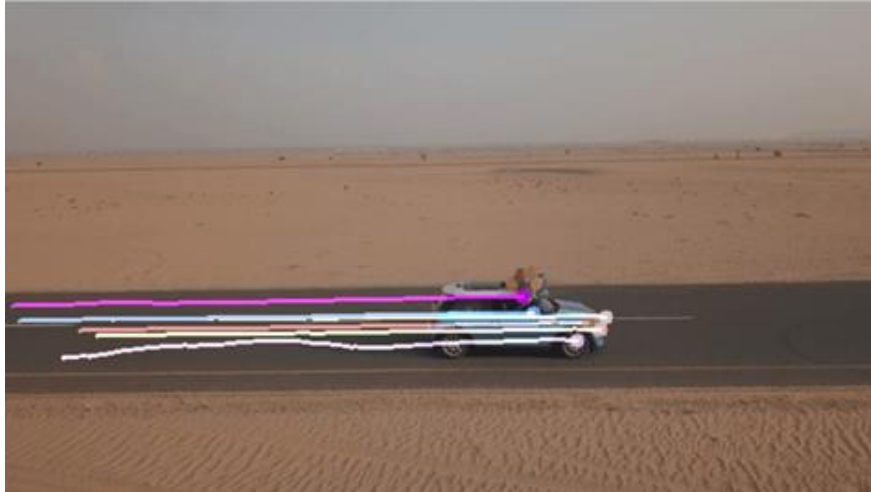


Рисунок 2.2 — Результат відслідковування за алгоритмом Optical flow [59]

Таким чином, алгоритм CAMShift підходить для задач із однорідним фоном і стабільним освітленням. У свою чергу алгоритм Optical Flow забезпечує більш високу точність і стійкість до шуму, але потребує більше обчислювальних ресурсів внаслідок того, що розраховуються додатково вектори руху для кожного пікселя зображення.

Обидва алгоритми можуть бути інтегровані шляхом впровадження в електронні системи безпеки, враховуючи особливості їх налаштування [60].

2.2 Алгоритм зменшення шуму на основі згорткових нейронних мереж

Проблема зменшення шуму під час обробки цифрових повнокольорових зображень для створення якісного мультимедійного контенту залишається однією з ключових у сучасному світі, особливо коли йдеться про ефективність функціонування мультимедійних систем технічного зору [30]. Фактично, шум,

його наявність та подальший аналіз можна спостерігати на всіх етапах роботи із зображеннями: від їхньої зйомки, оцифрування та передавання до процесу стиснення. На кожному з цих етапів, через особливості роботи пристроїв фіксації зображень, переведення їх у цифрову форму та застосування алгоритмів обробки, шум має тенденцію до накопичення [61].

У певних умовах надмірний рівень шуму може суттєво впливати на оцінку складових зображення. Наприклад, у медичній інженерії значний рівень шуму під час отримання низькоконтрастних зображень за допомогою комп'ютерної томографії може призводити до помилкової діагностики [62], [63]. Крім того, слід враховувати, що рентгенівське випромінювання спричиняє додаткові артефакти, які формують ефект «променевого зміцнення».

В роботі [57] визначено особливості використання згорткових нейронних мережі для вирішення задач лише відстеження об'єктів, і при цьому не робиться жодний наголос про те, що саме зображення фіксації може бути з певними вадами. В статті [61] наведено класичний метод виявлення країв на основі ентропії для зашумлених зображень, для якого запропоновано авторами провести кластерізацію порогів. Зокрема, пропонується поділ зображення на частини за окремими областями. На жаль, розглянутий алгоритм характеризується певними вимогами до розмірів цих порогів, які необхідно визначати кожного разу залежно від виду об'єктів на зображення. Така математична процедура лише ускладнює розв'язання задачі зменшення яскравісного шуму зображення.

В роботі [62] використано каскадні глибинні нейронні мережі для відновлення зображень, які отримано на основі засобів комп'ютерної томографії. На жаль, викладені положення в статті не дозволяють їх використати саме для задач зменшення яскравісного шуму, оскільки міститься лише оглядово по-кроковий алгоритм відновлення виключно томографічних зображень без деталізації, щодо особливостей навчання мережі, доступу до коду щодо модифікації самої архітектури нейронної мережі для вирішення інших прикладних задач.

В оглядовій статті [63], так само наявна медична специфіка щодо вибору сфери аналізу зображень і так само, як і в роботі [62], пропонується використовувати глибокі нейронні мережі. Водночас, для зменшення шуму в зображенні запропоновано метод візуалізації, який обмежується лише зображеннями, які отримано на основі приладів низькодозової комп'ютерної томографії. Крім цього, в статті розглянуто фреймворк, проте його практичне використання можливе лише, після придбання відповідної комерційної ліцензії.

Наявність шуму в супутникових зображеннях також є важливим фактором, що може призводити до втрати деталей під час розвідки місцевості. Дослідження, проведене у роботі [24], вказує, що супутникові радіолокаційні зображення із синтезованою апертурою є складними для аналізу через наявність спекл-шуму. Для забезпечення точної роботи радіолокаційних систем цей тип шуму необхідно обов'язково фільтрувати перед подальшою обробкою зображень.

У статті [25] досліджуються методи аналізу та відновлення зображень у дерматологічних застосуваннях. Автори наголошують, що одним із головних факторів, який впливає на якість зображення, є розмиття його елементів, що є різновидом шуму. Аналіз ядра розмитості дозволяє значно покращити роздільну здатність зображення та його якість. Додатково, у роботі [64] зазначено, що нехтування видаленням шуму може призвести до втрати корисної інформації. Для збереження країв та контурів об'єктів у зображенні пропонується застосування підходу анізотропної дифузії.

Дослідження [28] розглядає феномен «стего-шуму» та методи його зменшення. Стеганографічний аналіз є актуальною сферою, оскільки дозволяє створювати приховані повідомлення у різних видах аудіовізуального контенту, включаючи зображення.

Таким чином, проведений огляд літератури підтверджує актуальність проблеми зменшення шуму у цифрових зображеннях. У цьому розділі основна увага зосереджується на розгляді адитивного білого шуму, який найчастіше зустрічається при обробці цифрових зображень.

На відміну від зазначених статей [61-63] де переважно розглянуто глибокі нейронні мережі з певною специфікою підбору зображень, у цій роботі пропонується наскрізна глибока нейронна мережа згорткового типу для зменшення адитивного білого шуму на основі стратегії залишкового навчання. Відмінною особливістю роботи є детальний аналіз впливу різних рівнів шуму на ефективність нейронної мережі, що дозволяє адаптувати її до реальних застосувань у технічному зорі. На відміну від методів, що фокусуються на вузькому класі шумів або потребують значних обчислювальних ресурсів, запропонований підхід забезпечує високу продуктивність навіть у випадку обмежених обчислювальних можливостей. Крім того, отримані результати демонструють, що розроблена архітектура зберігає дрібні деталі зображень, не додаючи розмиття, що є важливим для систем відеоспостереження та технічного зору.

Таким чином, запропонований у цій роботі підхід дозволяє не лише підвищити ефективність зменшення шуму, а й зберегти якість оброблених зображень, що робить його практично значущим для широкого спектра застосувань.

При роботі з зображеннями, особливо коли вони фіксуються на основі зовнішньої камери відеоспостереження можуть бути ситуації, внаслідок несприятливої погоди на вулиці, коли картинка може бути не чіткою – підвищена вологість, дощ. Інша причина нечіткою картини зображення – вади роботи сенсорної світлочутливої матриці. У випадку, коли причиною є вади роботи сенсорної матриці камери, то на поверхні зображення може фіксуватися певна зернистість (шум зерна), яку при обробленні необхідно прибирати і відновлювати чіткість зображення, так нібито умови фіксації є ідеальними. Відомі різні алгоритми покращення зображення, з застосуванням програмних підходів і одним з них є використання глибинних нейронних мереж згорткового типу. В рамках даного дослідження необхідно було розробити метод видалення “яскравісного” шуму типу “зернистість” з зображень за допомогою глибинної згорткової нейронної мережі. Завдання полягало не лише в тому, щоб зменшити

шум, але й забезпечити збереження ключових деталей зображення, таких як краї, текстури та інші важливі ознаки [65].

Архітектура нейронної мережі, яка була створена у вигляді 40-згорткових шарів, поділена на три частини (рис.2.3):

- частина А – блок відокремлення ознак. Ця частина складається зі згорткового шару з 64 фільтрами, кожен із яких аналізує зображення на рівні текстурних ознак.

- частина Б – модуль з 16 залишкових блоків. Кожен блок включає два згорткових шари з функцією активації ReLU, що дозволяє врахувати нелінійність і зберегти ключові особливості зображення.

- частина В – блок відновлення. Використано вихідний згортковий шар із трьома фільтрами, який відновлює зображення до початкового кольорового формату. Основною задачею такою структури є насамперед виділення з зображення елементів шуму, паттерни якого потім подаються на елемент електронного суматора, який працює в режимі “віднімання”.

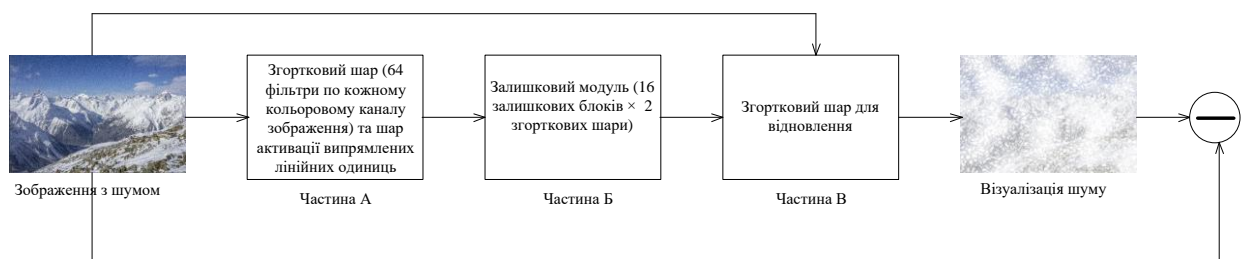


Рисунок 2.3 — Структурна схема нейронної мережі [65]

Для навчання мережі було обрано алгоритм оптимізації Адама зі швидкістю навчання 0,001. Було проведено 600 циклів навчання (епох). На основі функції втрат визначено динаміку зменшення похибки прогнозування нейронної мережі (рис.2.4) при навчанні. Значення функції втрат будемо визначати на основі такого співвідношення [66]:

$$Loss = \frac{1}{n} \sum_{i=1}^n \left(\frac{1}{w \cdot h} \sum_{j=1}^w \sum_{k=1}^h \|f_i(j,k) - X_i(j,k)\|^2 \right), \quad (2.1)$$

де f – зображення після очищення від шуму; X – оригінальне зображення без шуму; n – число семплів у кожному тренувальному пакеті; h – висота семплу; w – ширина семплу

Для навчання було направлено на вхід нейронної мережі набір 6862 зображень, де є елементи гір та сільської місцевості взимку [67].

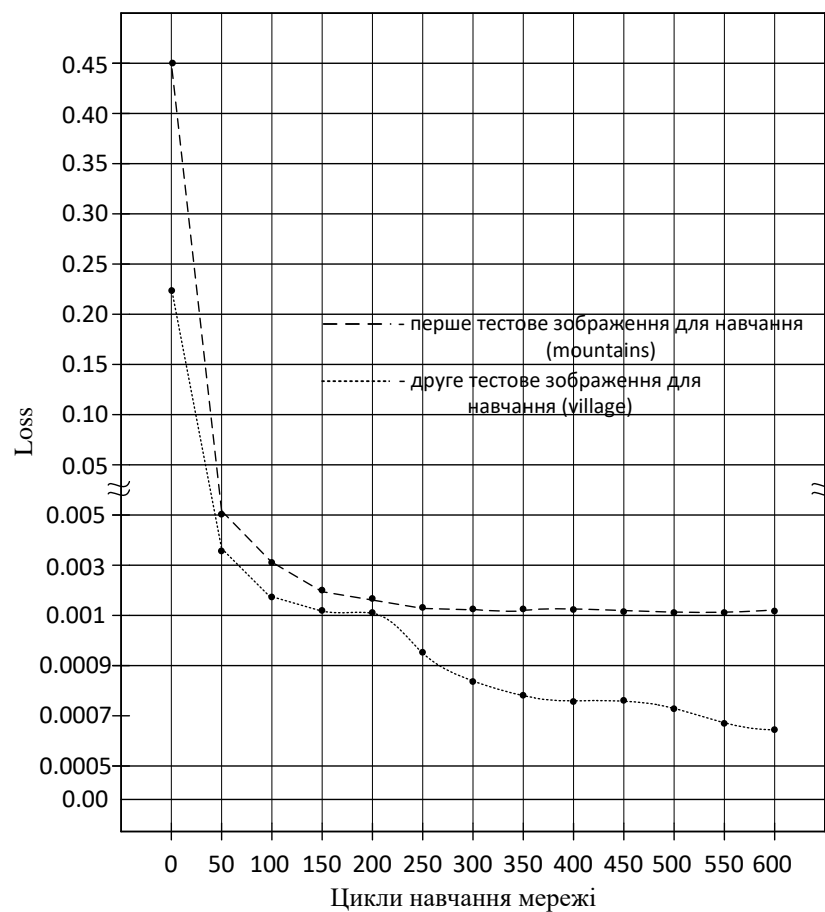


Рисунок 2.4 — Графічна залежність кривої сходження нейронної мережі [65]

Значення функції втрат (2.1) під час навчання нейронної мережі відображені у таблиці 2.1.

Таблиця 2.1 — Показники якості обробки зображень

Кількість циклів	1	100	200	300	400	500	600
<i>Loss_1</i>	0.45	0.003	0.00177	0.00141	0.00126	0.00119	0.00114
<i>Loss_2</i>	0.2205	0.00175	0.001045	0.00084	0.000758	0.000709	0.000674

Після навчання мережі була проведена перевірка, як розроблена схема дозволяє видалити зернистість на поверхні зображення. Для прикладу обрано тестове зображення, яке зроблено на фотоапарат, сенсорна матриця якого погіршує чіткість зображення вносячи невеличку зернистість. На рисунку 2.5 (б) наведено результат роботи нейронної мережі при значенні дисперсії шуму $\sigma=10$.



а)



б)

Рисунок 2.5 — Приклад роботи алгоритму з зменшення шуму зображення (дисперсія шуму дорівнює 10): а) - зображення з шумом; б) оброблене зображення після роботи нейронної мережі [65]

Додатково було розраховано метрики PSNR та SSIM та індекс структурної подібності SSIM для оцінки якості очищення зображення від шуму. (таблиця 2.2).

Метрика пікового відношення сигнал-шум визначається на основі такого співвідношення [30]:

$$PSNR(X, Y) = 10 \log \left(\frac{L^2}{\frac{1}{NM} \sum_{i,j} (X_{i,j} - Y_{i,j})^2} \right),$$

де X – оригінальне зображення; Y – отримане зображення зі зменшеним шумом як результат роботи нейронної мережі; L 255 динамічний діапазон зміни яскравостей для області розміром $N \times M$ пікселів зображення.

Індекс структурної подібності зображень можна визначити за формулою:

$$SSIM = \frac{(2m_x m_y + C_1)(2\sigma_{xy} + C_2)}{(m_x^2 + m_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)},$$

де $m_x m_y$, – математичне очікування зображень X та Y ; σ_x, σ_y – дисперсія зображень X та Y ; σ_{xy} – взаємна дисперсія зображень X та Y ; $C_1 = (k_1 L)^2$; $C_2 = (k_2 L)^2$; $k_1 = 0.01$; $k_2 = 0.03$.

Таблиця 2.2 — Показники якості обробки зображень

Дисперсія шуму	PSNR до (дБ)	PSNR після (дБ)	SSIM до	SSIM після
10	36,13	47,16	0,9652	0,9953
15	36,54	44,35	0,9534	0,9922
20	36,77	42,28	0,9427	0,9883
25	37,03	40,57	0,9256	0,9836

Аналіз отриманих даних показав, що за рахунок використання нейронної мережі згорткового типу метрика пікового відношення сигнал-шум PSNR після обробки на 4-5 дБ перевищує значення PSNR до проведення обробки на початковому рівні. Крім цього, розраховано, що індекс структурної подібності SSIM наближається до 1, що свідчить про збереження текстурних деталей.

2.3 Підходи до застосування алгоритмів вейвлет-перетворення для аналізу тексту на зображеннях

Для цифрового зображення, як однієї з ключових складових мультимедійного контенту, характерна широка сфера прикладних застосувань. Одним із таких напрямків є технологія розпізнавання та аналізу тексту на зображеннях, яка активно використовується в системах відеоаналітики. Зокрема, вона застосовується у комплексах аналізу текстового контенту в рекламних відеофайлах, у медичній сфері для розпізнавання тексту на зображеннях звітів, а також у інтелектуальних системах відеонагляду, що базуються на концепції інтернету речей. В основі цієї технології лежить використання різних методів, серед яких сегментація та підходи, що базуються на вейвлет-перетворенні. Це питання є предметом численних наукових досліджень.

У статті [68] вивчається технологія оборотного нанесення водяних знаків на зображення за допомогою вейвлет-перетворення під час створення текстових документів медичних звітів. Це перетворення використовується для захисту вмісту зображень та запобігання їх підробці. Автори пропонують метод шифрування знаків з використанням різних ключів, що дозволяє згодом виявляти та локалізувати змінені ділянки оригінального зображення. Щоб уникнути впливу підробки на вейвлет-область всього зображення, кожен блок перетворюється за допомогою вейвлет-перетворення в якості несучого сигналу для передачі інформації водяного знаку. На жаль, в роботі не розглянуто випадки, що стосуються розпізнавання та виділення текстової інформації, яка нанесена на складному оточуючому фоні зображення.

Цікаве застосування вейвлетів міститься в роботі [69], де запропоновано новий підхід до покращення якості зображень із надвисокою роздільною здатністю. Зокрема, автори пропонують використання глибокої згорткової нейронної мережі, що містить бета-вейвлет. Використання аналітичних фільтрів бета-вейвлет сприяє точнішій оцінці та відновленню втрачених

деталей при підвищенні роздільної здатності зображень. Проте, в статті [69] використані методи апроксимації та наближення, які не проаналізовано з огляду на оцінку точності та достовірності. До того ж, сама структура бета-вейвлету, його опис та відповідні бета-вейвлет-фільтри не наведено у відкритому доступі для їх безпосереднього використання при вирішенні різних практичних задач, які пов'язані з обробленням зображень.

Ще одне практичне застосування вейвлет-перетворення представлено в дослідженні [70], де для задач відеоаналізу та дактилоскопії запропоновано використання набору вейвлет-фільтрів на основі поліфазного представлення сигналу та алгоритму гравітаційного пошуку. Як приклад, у дослідженні використано вейвлет Даубекієса, а в результаті запропоновано метод проектування вейвлетів для обробки зображень. На жаль при цьому, наведений опис евристичного алгоритму гравітаційного пошуку неможливо використати для наших задач, так як, не наведено в статті які мають бути характеристики об'єктів на поверхні зображення.

Загальним питанням розпізнавання тексту на поверхні зображення присвячена робота [71]. У цьому дослідженні представлена гібридна модель класифікації відеозображень для підвищення ефективності процедур виявлення та розпізнавання тексту на зображенні. З цією метою застосовано модель ResNet50 для оцінки розподілу пікселів та мережу VGG16 для аналізу максимально стабільних екстремальних областей. Додатково використано каскадну згорткову нейронну мережу. Аналіз тестових зображень підтвердив зв'язок між якістю розпізнавання тексту та рівнем шуму у зображеннях. Проте, отримані результати в статті, не дивлячись на значну кількість наведеної візуальної інформації, на жаль важко використати для практики внаслідок того, що не наведено розроблений фреймворк.

Огляд методів детектування тексту на поверхні зображення наведено у роботі [72]. Автори підкреслюють важливість виділення семантичної інформації з кадру та наголошують, що виявлення тексту у відеопослідовності є складним технічним завданням. Фактори, що можуть ускладнити

розпізнавання, включають: низьку роздільну здатність, складний фон, нерівномірний текст, а також зміни його розміру, кольору та стилю. Проте у роботі не представлено узагальнену систематизацію обмежень методів, що базуються на вейвлет-функціях.

Огляд літературних джерел [68-72] підтвердив актуальність використання вейвлет-перетворень у завданнях обробки зображень. Водночас у розглянутих дослідженнях не запропоновано ефективних алгоритмів розпізнавання тексту без застосування нейронних мереж. Це створює необхідність розроблення простого програмного алгоритму виділення тексту із зображення, що характеризується швидким аналізом та високою точністю розпізнавання.

На відміну від зазначених робіт, у цій роботі представлено програмний алгоритм розпізнавання тексту на поверхні зображення, який базується на вейвлет-перетворенні та використанні вбудованих функцій у бібліотеках середовища PyCharm. Основною перевагою запропонованого підходу є його висока швидкість аналізу та можливість роботи в умовах обмежених обчислювальних ресурсів.

Важливим аспектом цієї роботи є також оцінка недоліків розробленого алгоритму, а саме його обмежена здатність коректно розпізнавати символи нелатинського алфавіту через особливості роботи OCR-модуля. Однак запропонований підхід забезпечує достатньо високу точність розпізнавання латинських.

В деяких практичних ситуаціях при роботі, наприклад, електронних систем безпеки дорожнього руху виникає необхідність, окрім фіксації об'єктів (автомобілі, пішоходи), визначення їх приналежності за певними ознаками, проводити зчитування і текстової інформації на цих об'єктах. Одним з таких прикладів може бути зчитування номерних знаків автомобілів. В роботі розроблено алгоритм з використанням вейвлет-перетворення для вирішення задачі виділення та розпізнавання текстової інформації на поверхні динамічного зображення. Ключовою вимогою при цьому необхідно забезпечити саме точність визначення текстових символів у випадку, коли

оточуючий фон не є однорідними і має додаткові геометричні фігури (криві лінії), частина з яких може нагадувати текстові символи. В дослідженні використано вейвлет-функцію Хаара.

Якщо кольорове зображення представити у вигляді сигналу функції, то суть застосування вейвлет-перетворень полягає в розкладанні цього сигналу за базисом, утвореним шляхом зсувів і масштабованих копій функції-прототипу. Такі базисні функції називаються вейвлетами, за умови, що вони визначені у просторі $L^2(\mathbb{R})$ (де $L^2(\mathbb{R})$ — простір комплексно значущих функцій $f(x)$ на прямій із обмеженою енергією), а також характеризуються коливанням навколо осі абсцис і швидким спаданням до нуля зі збільшенням абсолютного значення аргументу [16].

Процедура згортки сигналу з вейвлет-функцією дає змогу визначити характерні особливості сигналу в околі локалізації вейвлета. Базисну функцію перетворення можна записати наступним чином [16], [73]:

$$F_{a,b} = \frac{1}{\sqrt{a}} F\left(\frac{x-b}{a}\right), a \in \mathbb{R}^+, b \in \mathbb{R};$$

тоді вейвлет-перетворення матиме вигляд

$$W(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} F\left(\frac{x-b}{a}\right) \cdot f(x) dx.$$

Параметр b визначає розташування вейвлета в часі, a — параметр масштабу. Щоб можна було отримати зворотне вейвлет-перетворення, базисна функція $F(x)$ - має задовольняти енергетичній умові:

$$\int_0^{\infty} \frac{|\Phi(\omega)|^2}{\omega} d\omega < \infty,$$

де $\Phi(\omega)$ - перетворення Фур'є від функції $F(x)$.

Для побудови алгоритму розпізнавання тексту на поверхні зображення використаємо один із найпростіших вейвлетів – вейвлет Хаара (рис. 2.6) [18], [74].

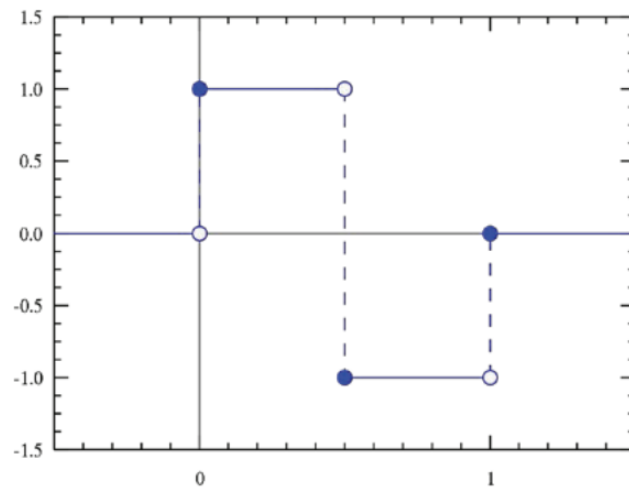


Рисунок 2.6 — Графічне зображення функції вейвлета Хаара [74]

Розроблено алгоритм, який визначається наступними етапами: завантаження зображення та переведення його в чорно-білий формат за допомогою порогової обрізки (метод Оцца); застосування вейвлет-перетворення Хаара для виділення ключових ознак тексту; реконструкція зображення та видалення фонових елементів; оптичне розпізнавання тексту за допомогою бібліотеки pytesseract, яке доступно в середовищі об'єктно-орієнтованого програмування pyCharm. Блок-схема алгоритму виділення та розпізнавання тексту показана на рисунку 2.7 а сам лістинг програми наведено у додатку А роботи [75]. Таким чином, реалізовано на основі команд та бібліотек порогову обрізку для виділення тексту на зображенні, використано метод вейвлет-перетворення Хаара для отримання частотних коефіцієнтів, і проведено реконструкцію зображення з текстом із подальшим застосуванням оптичного розпізнавання символів за допомогою бібліотеки pytesseract. В якості обрано зображення з текстом, яке наведено на рисунку 2.8.

На рисунку 2.9 представлено результат роботи алгоритму з виділення та розпізнавання тексту зображення зі складним оточуючим фоном в терміналі середовища pyCharm. Разом тим, знайдено певні особливості роботи створеного алгоритму. По-перше, не дивлячись на те, що оригінальне зображення є кольоровим, сам алгоритм працює з зображенням, як з чорно-білим.

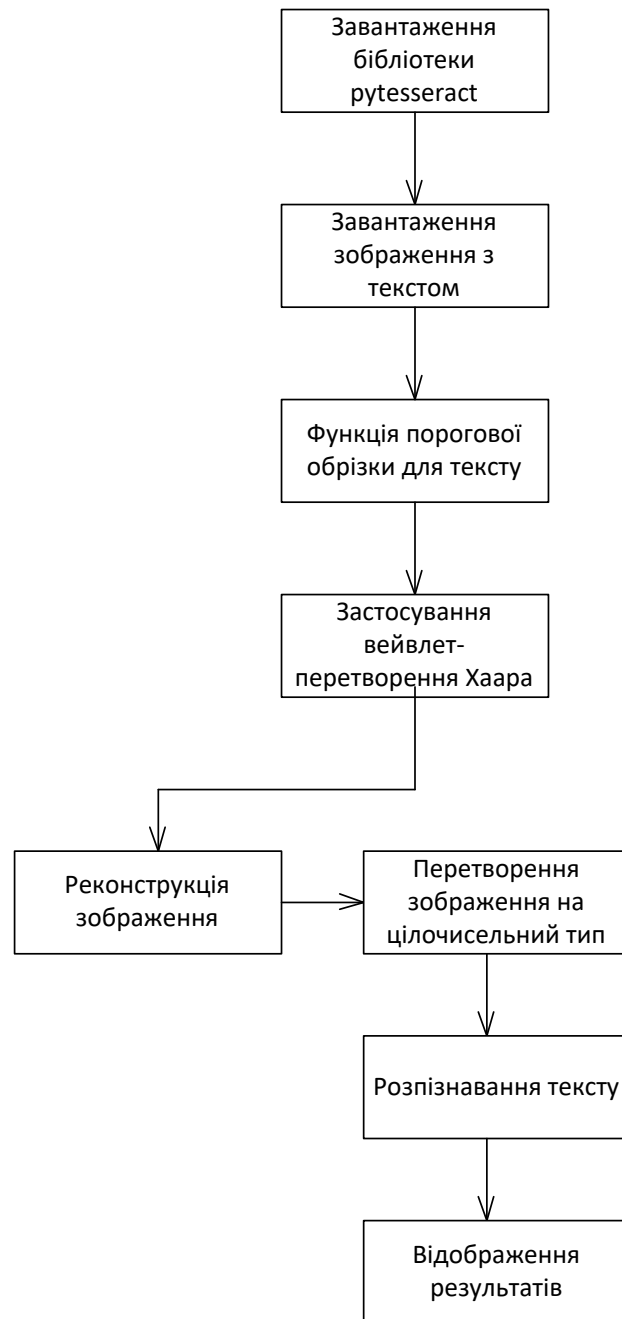


Рисунок 2.7 — Структурна схема алгоритму виділення та розпізнавання тексту [75]

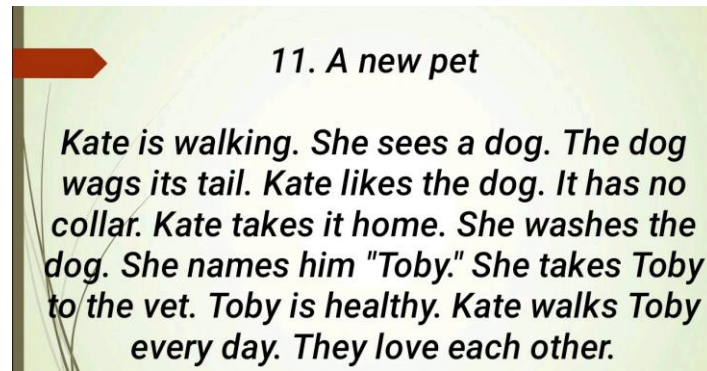


Рисунок 2.8 — Тестове зображення з текстом

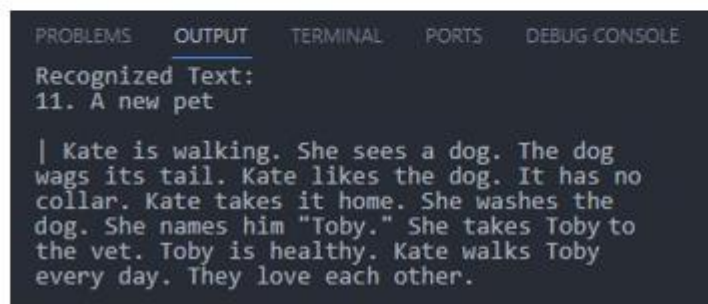
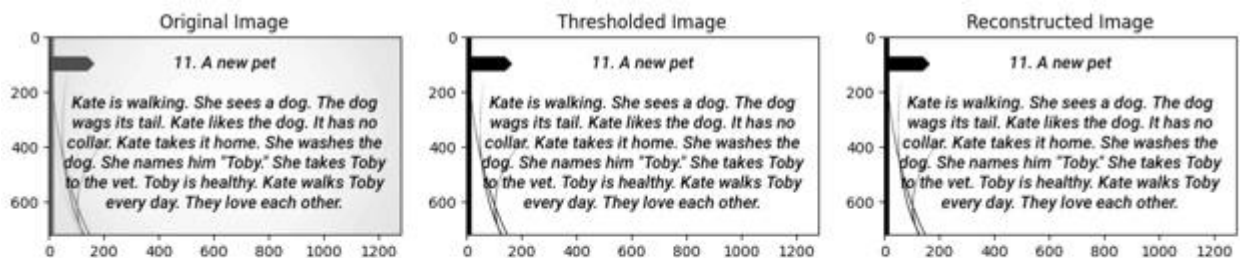


Рисунок 2.9 — Результати роботи алгоритму [75]

По-друге, алгоритм розпізнавання тексту є стійкий до фонових елементів у формі кривих ліній. Тобто, обрамлення зображення має рисунок у формі кривих ліній, які перетинають в оригінальному зображенні слова “dog”, “collar” та “to”. Як можна переконатись на отриманому результаті, алгоритм не сприймає ці додаткові елементи і не спотворює відтворений текст (рис.2.9). Можна передбачити, що розпізнавання символів проходить циклічні етапи перевірки з відкиданням зайвих елементів зображення. На рисунку 2.10 наведено ще один приклад перевірки розпізнавання тексту в зображенні.

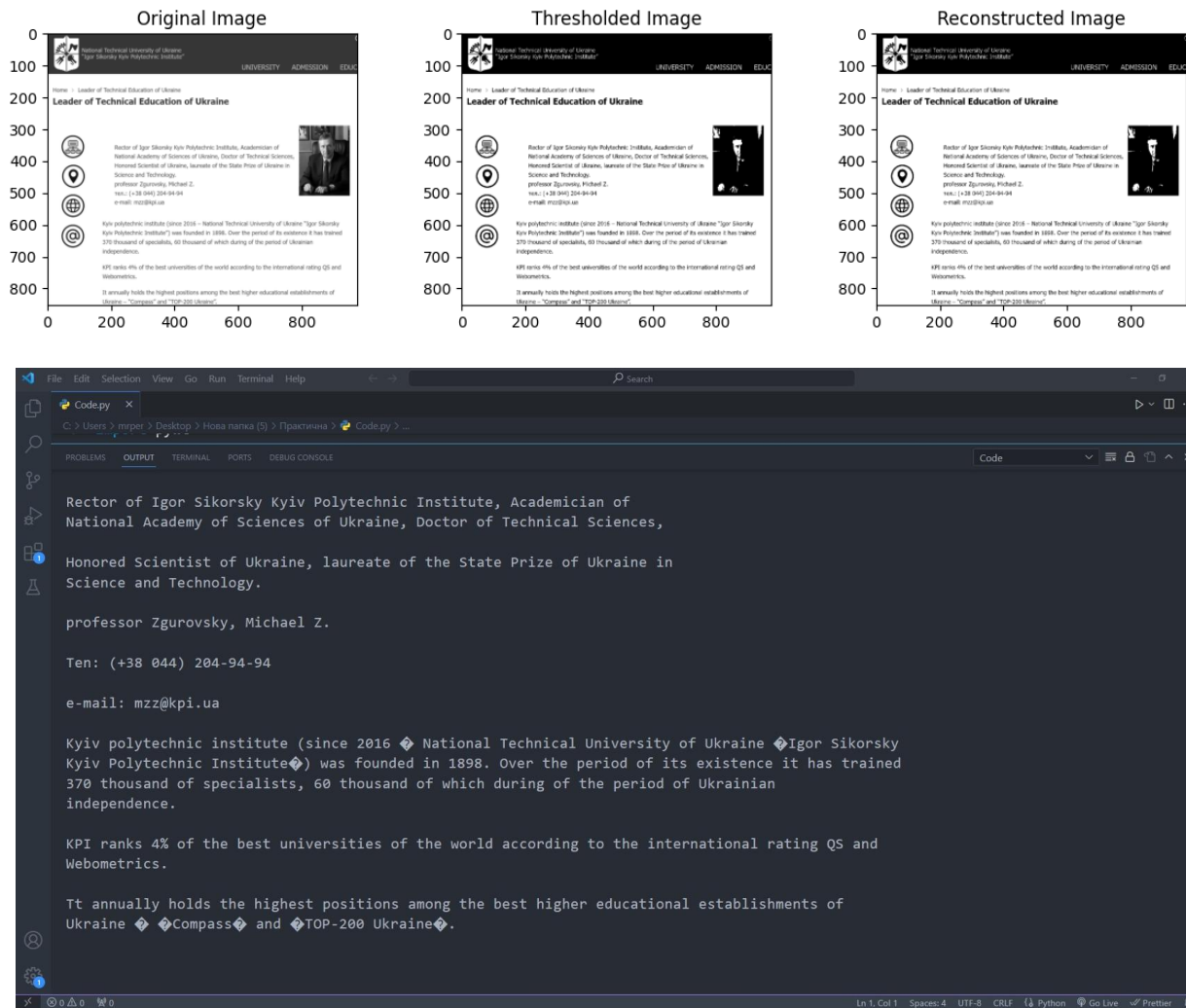


Рисунок 2.10 — Результати роботи алгоритму

З рисунку 2.10 можна відмітити, що алгоритм не коректно працює з деякими символами, насамперед це стосується символів типу “лапки”. Крім цього, створений алгоритм має і свої обмеження. Насамперед це стосується того, що він не коректно проводить розпізнавання літер не латинського алфавіту. Вочевидь, це обумовлено особливостями роботи модуля оптичного розпізнавання за допомогою бібліотеки pytesseract.

2.4 Експериментальні дослідження різних модифікацій моделі YOLO v7

Задачі виявлення об'єктів на поверхні зображення застосовуються у багатьох прикладних системах різних сфер науки і техніки. Крім того, виявлення об'єктів є частиною підходів до розпізнавання зображень, що дозволяє швидко і точно ідентифікувати об'єкти на зображенні. Наприклад, у системах відеоспостереження за громадським порядком у мегаполісах використовується програмне забезпечення для розпізнавання обличчя людей. У цьому випадку застосовуються методи виявлення об'єкта, тобто людини, пошук відповідного обличчя в базі даних та виведення результатів аналізу на зовнішні периферійні пристрої. Іншим прикладом застосування алгоритмів виявлення об'єктів є безпілотні транспортні засоби та інші інтелектуальні транспортні системи [35]. У цьому випадку такі алгоритми використовуються для виявлення пішоходів, зчитування дорожніх знаків [38], ідентифікації інших транспортних засобів на дорозі. У роботі [76] розглянуто програмний алгоритм ES-YOLO для виявлення втоми водія у режимі реального часу. Запропонований підхід оптимізує базову модель YOLO v7 шляхом модифікації "блоку уваги" у структурі відповідної згорткової нейронної мережі.

Цікаве застосування алгоритмів виявлення об'єктів відзначається у сфері інфраструктурного моніторингу, зокрема при аналізі дорожніх мостів для раннього виявлення тріщин і дефектів поверхні. Крім того, ці алгоритми використовуються при розробці систем для виявлення витоків у резервуарах, що зберігають небезпечні хімічні речовини.

Окремим напрямком застосування моделей виявлення об'єктів є їх використання у комп'ютерному аналізі зображень для вирішення різних прикладних задач. Наприклад, такі моделі використовуються у програмному забезпеченні для комп'ютерної скелетної анімації як основа для ріггінгу у 3D-графіці. Крім цього, вони застосовуються у створенні тизерів, оцінці дорожньої чи супутникової місцевості з формуванням карт глибини. Додатковим

напрямок є використання вдосконалених алгоритмів YOLO v7 у роботі безпілотних літальних апаратів [36], [51], [77], особливо при аерофотозйомці дронів [51] для швидкого розпізнавання лісових пожеж у реальному часі [44] або, навпаки, наземного спостереження контрольованих зон аеропортів [78] чи місцевості [35], [79], [80].

На відміну від зазначених досліджень, у цій роботі запропоновано реалізацію алгоритму YOLOv7 із підключенням додаткових модулів уваги та оптимізацією процесу навчання.

Модель YOLO за своєю архітектурою характеризується тим, що в її структурі використовується лише одна нейронна мережа згорткового типу FasterNet-L і особливість роботи моделі полягає у тому, що аналіз медіа контенту проходить лише один раз і в один прохід. При цьому нейронна мережа обов'язково має бути проведена крізь етап навчання на основі визначеної бази даних. В роботі [81]. Для навчання мережі використано базу даних MS COCO, яка містить близько 123287 зображень у 80 категоріях. Для окремих модифікацій моделі YOLO v7 додатково використано базу даних EuroCity, яка містить зображення об'єктів міської інфраструктури. Крім цього, в структурі роботи моделі підключено додаткову мережу ознак FPN, яка використовується для поєднання точної інформації при місцезнаходженні об'єкту з неглибоких шарів нейронної мережі з семантичною інформацією з глибоких шарів нейронної мережі. Враховуючи, що самі складові мультимедійного контенту (зображення, відео), які може записувати камера відеонагляду можуть характеризуватись розмиттям елементів, наявністю маленьких деталей на поверхні зображення, то на перший план виступає саме швидкість аналізу зображень в режимі реального часу та точність визначення ознак об'єктів. З метою перевірки цієї точності при роботі моделі YOLO v7 в рамках практичного експерименту, відібрано десять різних зображень, які послідовно проведено через нейронну мережу згорткового типу. Наприклад, на рисунку 2.10 наведено одне з набору тестових зображень (3640×2100; 7.3 Мп; DPI: 72 ppi; RGB 8 біт на канал).

Попередньо, нейронна мережа з підключеною функцією активації LeakyReLU проведена через процедуру навчання. Причому, в рамках експерименту проаналізовано 6 модифікацій моделі YOLO v7, а саме: v7 (базовий алгоритм); v7x (більш складна архітектура нейронної мережі та додано нову функцію активації Mish); w6 (6 шарів згортки); e6 (при навчанні мережі використано набір даних EuroCity); d6 (6 шарів згортки та використано удосконалений механізм уваги для виявлення дрібних деталей та об'єктів з складною конструкцією); e6e (для навчання мережі використано набори даних відразу з двох баз зображень COCO та Eurocity).



Рисунок 2.11 - Одне з тестових зображень

Для цього зображення як і для іншого тестового набору було мережею класи для аналізу. Так, для рисунку 2.11 це: людина; автомобіль; автобус; вантажівка; світлофор.

В результаті роботи моделі отримано дані про сам об'єкт, а саме клас об'єкту, який був призначений нейронної мережею, на основі попереднього навчання; центр області об'єкту (нормовані координати центру об'єкту x та y відносно ширини та висоти зображення); ширина та висота об'єкту (нормовані значення відносно ширини та висоти зображення). На рисунку 2.12 наведено частину цих числових даних для зображення з рисунку 2.11.


```

1.txt
1 2 0.242582 0.378607 0.0505495 0.0656716
2 2 0.171429 0.360199 0.0340659 0.0358209
3 7 0.213187 0.362189 0.0626374 0.0776119
4 2 0.213874 0.360697 0.0607143 0.0776119
5 0 0.172802 0.472637 0.0291209 0.0567164
6 5 0.126236 0.313184 0.0376374 0.0562189
7 7 0.0123626 0.26592 0.0241758 0.199502
8 2 0.116346 0.417164 0.053022 0.0930348
9 9 0.0879121 0.132836 0.00989811 0.0477612
10 2 0.0953297 0.376368 0.0335165 0.0731343
11 2 0.312088 0.412687 0.119231 0.128856
12 2 0.44739 0.441542 0.0898352 0.127861
13 2 0.134478 0.441045 0.0744506 0.123881
14 9 0.281593 0.163184 0.00934066 0.0427861
15 0 0.0546703 0.350746 0.0192388 0.0636816
16 2 0.588736 0.489303 0.157692 0.11592
17 2 0.34739 0.451493 0.134341 0.129851
18 0 0.84794 0.451741 0.0260989 0.120398
19 2 0.716621 0.473383 0.149725 0.0830846
20 0 0.763324 0.418657 0.0217033 0.0681592
21 2 0.529945 0.555721 0.176374 0.216915
22 2 0.600275 0.427363 0.185714 0.0915423
23 2 0.7 0.671393 0.408791 0.238308
24 2 0.190247 0.528358 0.14478 0.261692
25 2 0.920055 0.583333 0.140659 0.221393
26 2 0.794643 0.596269 0.24533 0.216418
27 2 0.670055 0.873881 0.654396 0.240299
28 2 0.362363 0.682587 0.261538 0.383085

```

Рисунок 2.12 – Числові дані відпрацювання моделі

На рисунку 2.13 показано результат роботи моделі YOLO v7 з модифікацією w6 для зображення з рисунку 2.11.



Рисунок 2.13 - Візуалізація результатів роботи YOLO v7 w6 для виявлення транспортних засобів на зображенні [81]

В рамках практичного експерименту визначено для кожного з 10 зображень один з класів об'єктів і проаналізовано, які значення довірчих оцінок характерні для цього класу [81]. В таблиці 2.3 наведено, один клас об'єктів з відповідними обчисленими значеннями довірчих оцінок (наскільки ймовірно граничні рамки області зображення містять визначений клас об'єкту) для

зображення з рисунку 2.11. Вибір таких класів об'єктів на зображеннях обумовлений тим, що потрібно перевірити, чи можна використовувати розглянуті моделі, наприклад, в системах автопілоту транспортних засобів. Тобто, як точно дана модель YOLO функціонує і визначає дорожні знаки, світлофори, насичений місцевий трафік в місті, тощо. Друга задача, яка пов'язана з таким вибором обумовлена перевіркою моделей на точність їх роботи, коли на площині зображення є маленькі деталі, наприклад секція світлофору, або коли є перетин різних класів об'єкту.

У моделі YOLO v7 для оцінки ефективності виявлення об'єктів використовується метрика середньої точності (Average Precision, AP). AP є ключовим показником, який відображає здатність моделі точно ідентифікувати та локалізувати об'єкти на зображенні. Вона розраховується як площа під кривою залежності точності (precision) від повноти (recall) для конкретного класу об'єктів. В загальному випадку, цей параметр можна визначити за наступним співвідношенням [82]:

$$AP = \int_0^1 p(r)dr, \quad (2.2)$$

де $p(r)$ — функція залежності точності від повноти. У практичних застосуваннях інтеграл зазвичай обчислюється як сума значень точності при різних рівнях повноти. Середня AP для всіх класів, відома як mean Average Precision (mAP), є загальноприйнятою метрикою для оцінки продуктивності моделей виявлення об'єктів (табл.2.3) і її визначають так:

$$mAP = \frac{TP}{TP + FP} \times 100\%, \quad (2.3)$$

де TP – кількість випадків, коли впевненість моделі $\geq 0,5$; FP – кількість випадків, коли модель неправильно визначає тип об'єкту.

Таким чином, метрика meanAP дозволяє оцінити, наскільки добре модель виявлення об'єктів здатна правильно ідентифікувати та локалізувати різні об'єкти на зображеннях.

Таблиця 2.3 — Показники якості обробки зображень

Зображення	Обраний клас	Модель YOLO v7 та її модифікації, обчислене середнє значення mAP , %					
		v7 (базова)	d6	e6	e6e	w6	v7x
1	Віддалена поліцейська машина з мигалками	30	41	36	0	35	54

В роботі [81] авторами прийнято, що модель забезпечує достатню точність, якщо обчислене значення перевищує поріг 0,897 або 89,7%. Такий критерій обрано з розрахунком величини ймовірності, яке визначено розробниками для виявлення об'єктів на поверхні зображення, коли використовується для аналізу модель наступного покоління YOLO v8. Враховуючи це, можна переконатись що в умовах наднасиченої кількості об'єктів модель функціонує так, що не відповідає встановленому порогу, а тому в рамках дослідження для розв'язання наступної задачі, яка пов'язана з запровадженням нового класу об'єктів для динамічного зображення (визначення об'єкту зброї в руках зловмисника), яке фіксується за різних умов зйомки, в базову структуру нейронної мережі моделі YOLO v7 введено наступні зміни:

- збільшено глибину мережі (параметр depth multiple збільшено з 0,33 до 0,5);
- збільшено ширину мережі (параметр width multiple збільшено з 0,5 до 0,75);
- додано модуль (програмний механізм) уваги CBAM;
- додано 81 клас до стандартної класифікації моделі COCO і оновлено головні шари нейронної мережі для роботи з новим класом;
- створено спеціалізований набір даних (датасет) для донавчання нейронної мережі.

Порівнюючи отримані результати з роботи авторів [81], наприклад з аналогічним проведенням дослідженням, яке показано в статтях [37,46,57]

можна відмітити таке: показники *mAP* визначено для наймолодших моделей YOLO та YOLO v2; розглянута візуальна інформація є простою, насамперед тим, що об'єктів на площині зображення не більше шести (зображення з винищувачами) і вони між собою не перетинаються; для навчання нейронної мережі використано датасет від Оксфордського університета PASCAL VOC, підтримка та оновлення якого зупинилась у 2015 році.

Висновки до розділу 2.

1. Алгоритм Optical Flow є чутливим до змін форми об'єкта та потребує точного налаштування параметрів, таких як розмір вікна спостереження. Недоліком алгоритму є зниження точності при зменшенні об'єкта. Алгоритм CAMShift є простішим і менш вимогливим до параметрів, але його точність залежить від освітлення об'єктів, кольору та шуму. Optical Flow забезпечує глибший аналіз руху, проте складніший у налаштуванні. CAMShift підходить для швидкого відстеження в простих умовах, тоді як Optical Flow краще використовувати для точного аналізу в складних сценаріях.

2. Застосована згорткова нейронна мережа ефективно зменшує зернистість зображень, зберігаючи важливі деталі, такі як краї та текстури. Навчання на 6862 зображеннях із використанням алгоритму Adam зі швидкістю 0.001 протягом 600 епох дозволило знизити похибку прогнозування. Показник PSNR після обробки зріс на 4-5 дБ, наприклад, при дисперсії шуму 10: з 36,13 дБ до 47,16 дБ. Індекс структурної подібності (SSIM) збільшився з 0,9652 до 0,9953 при дисперсії шуму 10, що свідчить про високу якість збереження текстур. Таким чином, згорткові нейронні мережі є ефективним інструментом для очищення зображень від шуму.

3. Алгоритм на основі вейвлет-перетворення Хаара ефективно розпізнає текст на зображеннях зі складним фоном, ігноруючи фонові елементи. Він забезпечує точне розпізнавання тексту, проте має обмеження при обробці символів не латинського алфавіту та спеціальних знаків, таких як лапки. Причиною є особливості роботи бібліотеки pytesseract. Алгоритм є

перспективним для застосування, але потребує подальшого вдосконалення для розширення мовної підтримки.

4. Дослідження модифікацій YOLO v7 показали, що ефективність моделі залежить від її архітектури, набору даних і параметрів навчання. Модифікація v7x досягла найвищої довірчої оцінки 0,54 (у відсотках 54%), тоді як базова модель – 0,3 (у відсотках 30%). Версія d6 продемонструвала найкращу деталізацію об'єктів із показником 0,41 (у відсотках 41%), що робить її ефективною для складних умов. Модель w6, яка містить шість згорткових шарів, показала середній результат 0,35 (у відсотках 35%), забезпечуючи баланс між точністю та швидкістю. Вибір оптимальної модифікації залежить від конкретних завдань, таких як виявлення дрібних об'єктів або обробка відеопотоку в реальному часі.

5. Запропоновано модифіковану архітектуру моделі YOLO v7 з застосуванням ідеї використання капсульних мереж. Такий підхід дозволяє частково вирішити проблему перекриття об'єктів при роботі моделі YOLO v7.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ РОБОТИ АЛГОРИТМІВ ДЛЯ ЕЛЕКТРОННИХ СИСТЕМ БЕЗПЕКИ

У сучасних електронних системах безпеки широко застосовуються алгоритми виявлення об'єктів, засновані на нейронних мережах. Серед найбільш відомих моделей виділяється YOLO (You Only Look Once), яка забезпечує високу швидкість і точність виявлення об'єктів у реальному часі [7]. Однак існують і інші підходи, які мають свої переваги та недоліки.

Алгоритм Віола-Джонса є одним із перших методів виявлення об'єктів у реальному часі. Він використовує каскад простих функцій, схожих на Хаар, для швидкого виявлення обличчя на зображеннях. Перевагою цього методу є його швидкодія та відносна простота реалізації. Однак він обмежений у здатності виявляти об'єкти різних класів і чутливий до змін освітлення та положення об'єкта [83].

Згорткові нейронні мережі (CNN) стали основою для багатьох сучасних систем виявлення об'єктів. Вони здатні автоматично виділяти ознаки зображення та класифікувати об'єкти з високою точністю [34]. Перевагою CNN є їхня гнучкість і здатність навчатися на великих наборах даних. Однак для їх ефективної роботи потрібні значні обчислювальні ресурси та великі обсяги даних для навчання [48].

Рекурентні нейронні мережі (RNN) використовуються для обробки послідовних даних, таких як відеопотоки. Вони здатні враховувати часову залежність між кадрами, що дозволяє ефективно виявляти та відстежувати об'єкти в реальному часі. Перевагою RNN є їхня здатність працювати з послідовними даними, але вони можуть бути складними в навчанні та вимагати значних обчислювальних ресурсів [84].

Модель YOLO v7 поєднує в собі переваги високої швидкості та точності виявлення об'єктів. На відміну від алгоритму Віола-Джонса, YOLO здатна виявляти об'єкти різних класів і менш чутлива до змін умов зйомки [39]. Порівняно з традиційними CNN, YOLO є більш ефективною в реальному часі,

оскільки виконує виявлення та класифікацію об'єктів за один прохід мережі [85]. Хоча RNN добре справляються з послідовними даними, YOLO забезпечує кращу продуктивність для статичних зображень або окремих кадрів відео [86].

Таким чином, хоча існують різні підходи до виявлення об'єктів, модель YOLO v7 пропонує оптимальний баланс між швидкістю та точністю, що робить її ефективним вибором для сучасних електронних систем безпеки [87].

3.1 Розробка електронної системи безпеки

Для реалізації практичних алгоритмів з використанням нейронних моделей та власне моделі YOLO необхідно розробити типову електронну систему безпеки з вибором відповідного обладнання, яка б надала можливість користувачу провести відео аналіз зображення, як в режимі реального часу, так і для записаного відеосигналу з камер. Причому необхідно забезпечити можливість пересилання зображення з системи безпеки у хмарне середовище і віддалене керування як окремими камерами так і комутаторами з сервером. Більше того, обране обладнання має підтримувати певні інтелектуальні особливості, що стосуються об'єктів, які фіксуються камерою. Крім цього, сама система безпеки має характеризуватись тим, що в своїй структурі має ділянки з двома видами проводового зв'язку – волоконнооптичні лінії зв'язку та зв'язку з кабелем типу “кручена” пара. В першому випадку передбачається що є доступні LC-конектори, для другого – роз'єми типу RJ-45.

Далі наведемо особливості обраного обладнання для проектування системи безпеки і визначимо власне як адаптувати розроблені рішення з застосування нейронних мереж до функціонування цієї системи безпеки. В якості об'єкту, де розгорнута система безпеки обрано житловий комплекс де розташовані IP-камери як ззовні, так і всередині приміщення багатоквартирного будинку.

На рисунку 3.1 наведено типову схему електронної системи безпеки. В даному випадку передбачається підключення 6 IP-камер фірми Balter, 4

гібридних комутаторів фірми Versitron, 3 NVR-клієнтів що визначає собою персональний комп'ютер з клавіатурою, де встановлено спеціальне програмне забезпечення, 1 NVR-серверу фірми Ajax.

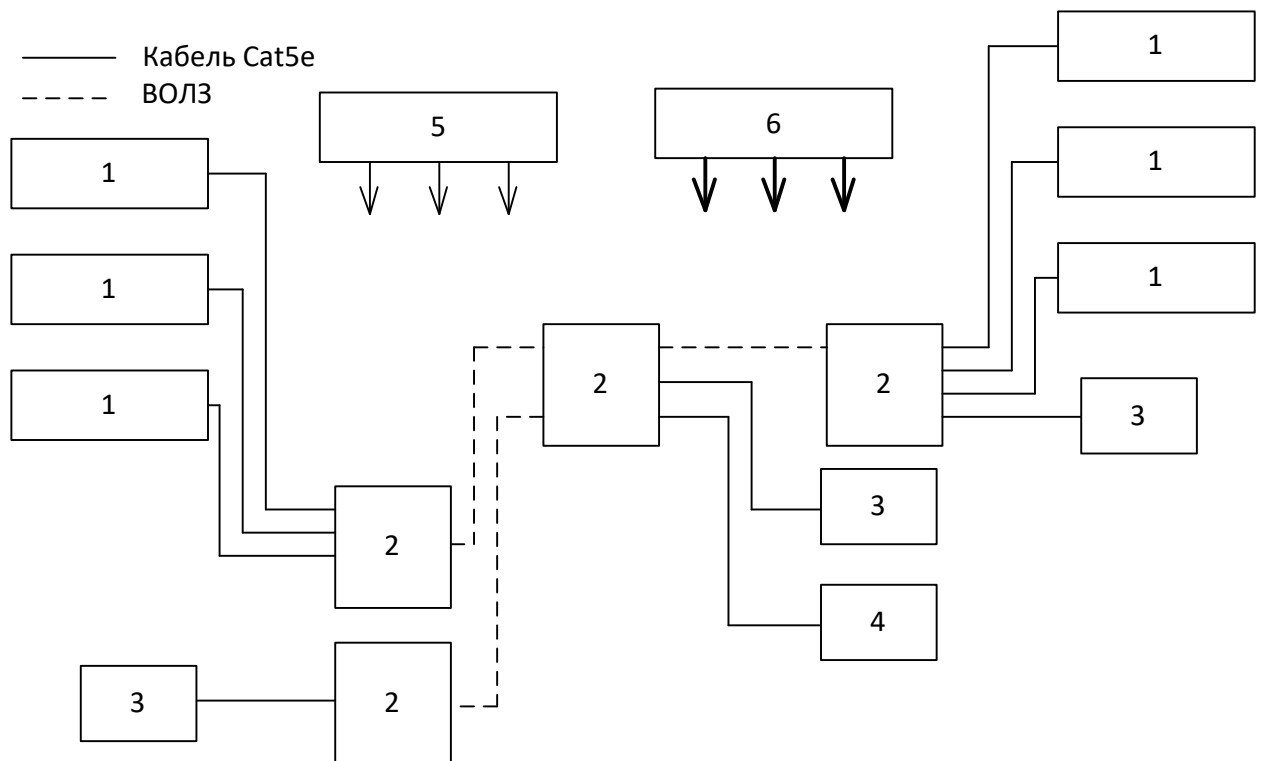


Рисунок 3.1 – Структурна схема електронної системи безпеки: 1 – IP-камера; 2 – гібридний оптичний комутатор; 3 – NVR-клієнт; 4- NVR-сервер; 5 – драйвер живлення 12 В; 6 – драйвер живлення для комутатора та сервера 110-240 В AC, 50-60 Гц, 1 А

В якості IP-камери візьмемо розробку фірми Balter моделі IP-MT1210GR [88], яка має інфрачервоне підсвічування, об'єктив 3,6 мм, розмір сенсорної матриці 2 МП. Зокрема, камера має інфрачервоні світлодіоди SMD з інноваційною технологією SMART IR і може виявляти об'єкти в режимі повної темряви. Передавання зображення з частотою 25 кадрів за секунду і впроваджено при передачі картинки технологію стиснення H.265. Корпус камери відповідає ступеню захисту IP66 і зроблено за алюмінію. Живлення камери можливе або через драйвер живлення 12 В DC або на основі технології POE. Для нашого дослідження обрано перший варіант.

Сигнал з камери надходить на гібридний оптичний комутатор, якості якого обрано розробку фірми Versitron моделі SG72860M [89]. Основні характеристики цього комутатора наведено у таблиці 3.1.

Таблиця 3.1 – Основні характеристики гібридного оптичного комутатора SG 72860M

Опис характеристики (параметра) комутатора	Значення
Можливість управління роботою та портами комутатора	Так
Загальна кількість портів	28
Кількість портів для підключення за RJ-45	24
Кількість портів для підключення ВОЛЗ	4
Загальна пропускна здатність	192 Гб/с
Розмір буферної пам'яті	4,1 МБ
Споживання, Вт	В режимі очікування 18 Вт, в режимі роботи 35 Вт

Відмітимо, що для обраного комутатора порти є гігабітними. Додатково є підтримка як одномодових (8/125 мм; 9/125 мм), так і багатомодових (50/125 мм; 62,5/125) оптичних волокон на основі використання SFP слотів з відповідним LC-з'єднувачем. Крім цього, є підтримка різних програмних інтересів та протоколів управління – Telnet, Web Management, тощо.

Також в схемі на рисунку 3.1 показано NVR-клієнт. Фактично це комп'ютерна система з встановленим спеціальним програмним забезпеченням для управління, запису та контролю за зображеннями з камер. Для системи безпеки обрано програмну розробку NUUO NVR IP+ (Тайвань) [90]. Програмні можливості розробки об'єднують IP-камери та NVR-сервер в єдину систему безпеки. Серед можливостей цього застосування - постійний запис або запис на

основі детектування руху, визначення незнайомого об'єкту чи загубленого, фіксація втрати сигналу з камери або втрата фокусу камери. Також є можливість керування рухом PTZ-камер та підрахувати кількість об'єктів на площині зображення з камери. Основним елементом електронної системи безпеки в схемі на рисунку 3.1 – NVR-сервер. Саме завдяки цьому елементу, який за призначенням ще називають, мережевий відеореєстратор є можливість проводити відеоаналіз зображення з камер, як в режимі реального часу, так і у випадку, вже записаного відеосигналу.

Для розробленої системи обрано розробку фірми Ajax NVR (16 ch) – мережевий відеореєстратор на 16 каналів [91] (рис.3.2).



Рисунок 3.2 – Зображення NVR-сервера

Основні характеристики серверу наведено в таблиці 3.2.

Таблиця 3.2 – Основні характеристики NVR-сервера Ajax NVR (16 ch)

Опис характеристики (параметра) комутатора	Значення
Вхідна пропускна здатність	100 Мб/с
Жорсткий диск, ємність	16 ТБ
Кількість IP-каналів за Ethernet	16
Технологія стиснення відео	H264, H265
Ступінь захисту	IP20
Підтримка зображення, роздільна здатність	до 4К включно
Швидкість запису	25 кадрів за секунду
Споживання, Вт	24

Даний елемент системи дозволяє проводити відеоаналіз зображення і сервер має опцію розпізнавання обличь, виявлення напрямку руху. Крім цього, є можливість додати підтримку додаткових функцій системи на основі завантаження програмного алгоритму та бібліотек, які дозволяють реалізувати нейронну мережу.

У порівнянні з іншими системами безпеки, такими як Hikvision [92] та Dahua [93], запропонована система, що складається з IP-камер Balter, гібридних комутаторів Versitron та NVR-сервера Ajax, має кілька переваг. Зокрема, використання камер Balter з інфрачервоним підсвічуванням SMART IR дозволяє отримувати якісне зображення навіть у повній темряві, що є перевагою перед деякими конкурентами [94]. Гібридні комутатори Versitron підтримують як мідні, так і оптичні з'єднання, що забезпечує більшу гнучкість у налаштуванні мережі порівняно з традиційними комутаторами [95]. Крім того, NVR-сервер Ajax підтримує розпізнавання обличь та аналіз напрямку руху, що підвищує ефективність моніторингу та реагування на інциденти [91]. Таким чином, запропонована система поєднує в собі передові технології та забезпечує високий рівень безпеки для житлового комплексу.

Можливі модифікації для покращення електронної системи безпеки.

У схемі використовується кабель Cat5e для з'єднання IP-камер із гібридними оптичними комутаторами. Це може бути вузьким місцем для передачі відео високої чіткості, особливо у разі роботи з великою кількістю потоків. Для підвищення швидкості та надійності передачі даних доцільно замінити Cat5e на Cat6 або Cat6a, що дозволить збільшити пропускну здатність і зменшити затримки у передачі сигналу.

Додатково можна впровадити комутатори з підтримкою PoE (Power over Ethernet), які дозволять одночасно передавати живлення та дані через той самий кабель. Це усуне потребу у зовнішніх блоках живлення для кожної камери та спростить монтаж. У схемі вже передбачено використання оптоволоконних каналів зв'язку (ВОЛЗ) між комутаторами, проте варто збільшити їхню кількість або модернізувати до сучасніших стандартів для підвищення стабільності з'єднань.

Щоб уникнути збоїв у разі відключення електроенергії, доцільно інтегрувати безперебійне джерело живлення (UPS), що забезпечить стабільну роботу всієї системи навіть при перебоях у мережі. Використання PoE-комутаторів також дозволить оптимізувати систему живлення, зменшивши кількість блоків живлення та підвищивши надійність системи.

Крім того, слід передбачити розподіл живлення на дві незалежні лінії (основна + резервна). Це дозволить у разі відмови одного джерела живлення переключитися на резервне, зберігаючи безперервну роботу всієї системи.

3.2 Додаткові можливості електронної системи безпеки

Використання нейронних мереж для автоматизованого виявлення зброї є одним із найбільш перспективних напрямків у галузі систем безпеки, особливо коли мова йде про умови, де видимість обмежена. У сучасному світі, де загрози безпеці стають все більш витонченими, традиційні методи виявлення зброї, що базуються на людському факторі або простих сенсорних пристроях, мають свої

обмеження. Часто, в складних умовах, таких як нічний час або оточуюче середовище з поганим освітленням, ефективність розпізнавання загроз значно знижується. Інфрачервоні камери вирішують цю проблему, дозволяючи фіксувати об'єкти навіть у відсутності видимого світла. Однак, для точного та швидкого розпізнавання об'єктів, таких як ножі чи інші види зброї, в реальному часі потрібна нейронна мережа, яка здатна з високою точністю обробляти ці зображення.

Розпізнавання зброї в умовах низької видимості пов'язане з багатьма труднощами. Наприклад, у складних умовах, таких як туман чи дим, видимість об'єкта може суттєво зменшуватись, що ускладнює його розпізнавання. По-друге, зображення, які отримано за допомогою інфрачервоних камер, часто містять шуми, які можуть впливати на точність виявлення. Це можуть бути відблиски або перешкоди від інших об'єктів у полі зору камери. Нейронна мережа повинна навчитися ігнорувати ці шуми і фокусуватися на об'єктах, що становлять потенційну загрозу. Крім того, зброя може бути частково прихована або триматися під різними кутами, що ускладнює її розпізнавання. Тому під час навчання нейронної мережі необхідно враховувати безліч ракурсів та положень об'єкта.

Ще однією критичною вимогою до такої системи, є її здатність діяти швидко та точно. У випадках, коли загроза може бути безпосередньою, система повинна реагувати в режимі реального часу. Це дозволяє службі безпеки негайно виявляти загрозу та оперативно реагувати, запобігаючи потенційним небезпекам. Модель YoloV7 є одним при цьому з можливих алгоритмів виявлення об'єктів, який обрано в роботі для вирішення задачі розпізнавання холодної зброї.

Традиційні технології виявлення, такі як металодетектори та звичайні камери, мають свої обмеження. Вони не завжди ефективні в умовах поганої видимості або на великих відстанях. Натомість інфрачервоні камери у поєднанні з передовими алгоритмами глибокого навчання, як YoloV7, можуть адаптуватися до різних умов оточуючого середовища, наприклад при наявності

несприятливих погодних умов на вулиці або при розпилюванні газу в приміщенні.

Таким чином, створення нейронної мережі з врахуванням змін, які наведено в п.2.4 роботи, для виявлення зброї за допомогою інфрачервоної камери в умовах низької видимості є важливою задачею при функціонуванні сучасних системи безпеки.

3.3 Формування спеціалізованого датасету для навчання нейронної мережі

У першому етапі експерименту було важливо створити якісний та об'ємний датасет, який максимально точно відображає різноманітність ситуацій, в яких можна виявити зброю в реальних умовах. Детальний код завантаження та підготовки датасету наведено в додатку Б, пункт 1. Цей процес вимагав ретельної підготовки та дотримання декількох принципів для забезпечення надійності та точності результатів.

Для збору матеріалів були використані різні джерела зображень ножів, а також власні фотографії. Основна увага була зосереджена на забезпеченні якомога більшої різноманітності зображень, щоб модель могла добре виявляти необхідні об'єкти. Характерні особливості, які були враховані при формуванні набору зображень для навчання нейронної мережі моделі YOLO v7:

- **Фото з різних ракурсів:** Зображення ножів були зроблені з численних ракурсів, таких як верхній вид, бічний вид, кут знизу і фронтальний вид. Це дозволило створити умови для розпізнавання об'єкта в складних і нестандартних положеннях, які можуть виникати в реальних ситуаціях.
- **Варіації освітлення:** Щоб нейронна мережа могла розпізнавати ножі при різних умовах освітлення, деякі зображення були зроблені при слабкому освітленні, з яскравими відблисками чи затіненими ділянками, що симулювало умови з недостатньою видимістю.

- **Різні фони:** Усі фото були зроблені на різних фонах: як на однорідному, так і на складних, з об'єктами в задньому плані, що додатково ускладнювало розпізнавання об'єкта.

Приклад зображень ножів для тренування на рисунку 3.3.

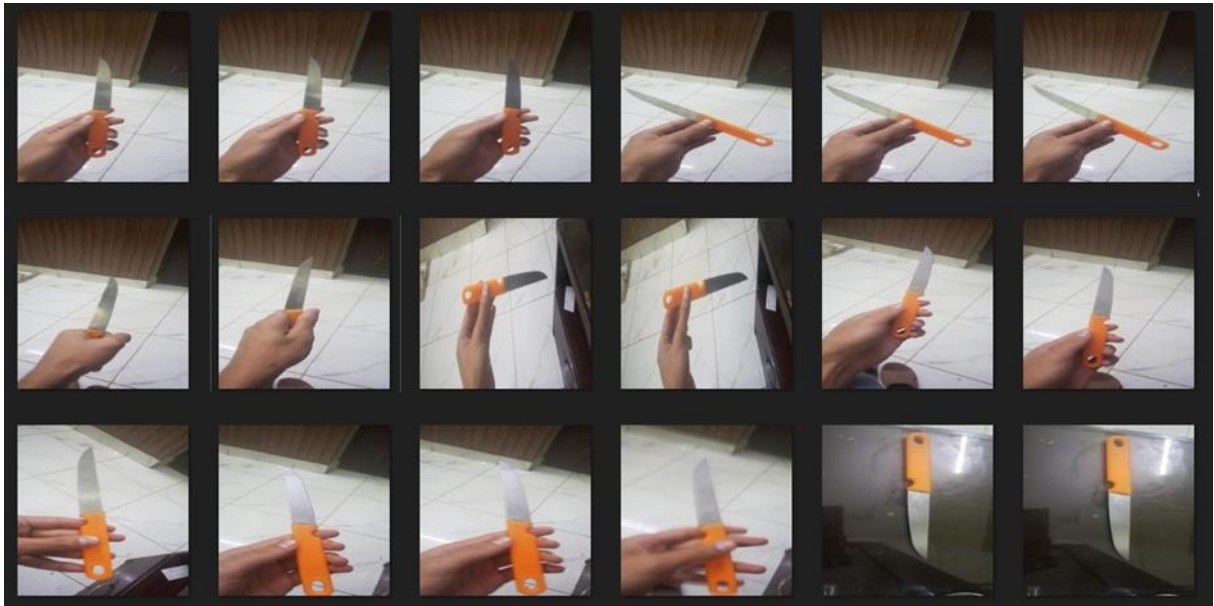


Рисунок 3.3 — Зображення ножів для тренування моделі: різноманітні ракурси, освітлення та фони

Щоб збільшити кількість зображень та забезпечити кращу продуктивність моделі, було використано різні методи розширення даних:

- **Дзеркальне відображення:** Кожне зображення було оброблене, щоб створити його дзеркальну копію. Це дозволило моделі бути стійкою до змін орієнтації об'єкта.
- **Масштабування:** Зображення ножів були змінені у розмірі для створення випадкових варіацій у вигляді великого або малого об'єкта, що навчило модель розпізнавати ножі на різних дистанціях.
- **Ротація та зсув:** Ножі були повернені під різними кутами, щоб модель навчилася коректно розпізнавати об'єкт незалежно від його положення. Також були додані випадкові зсуви по осі X та Y , щоб додати до зображень асиметрії.

- **Зміна яскравості та контрасту:** Деякі зображення були оброблені для зміни рівнів яскравості та контрасту, імітуючи різні умови освітлення, зокрема слабе або надмірне освітлення.
- **Шум та розмиття:** У деякі фото було додано шум або розмиття для симуляції ситуацій з поганою видимістю або наявністю перешкод, як-от пил чи дим.

Приклади зображень для розширення датасету на рисунку 3.4.

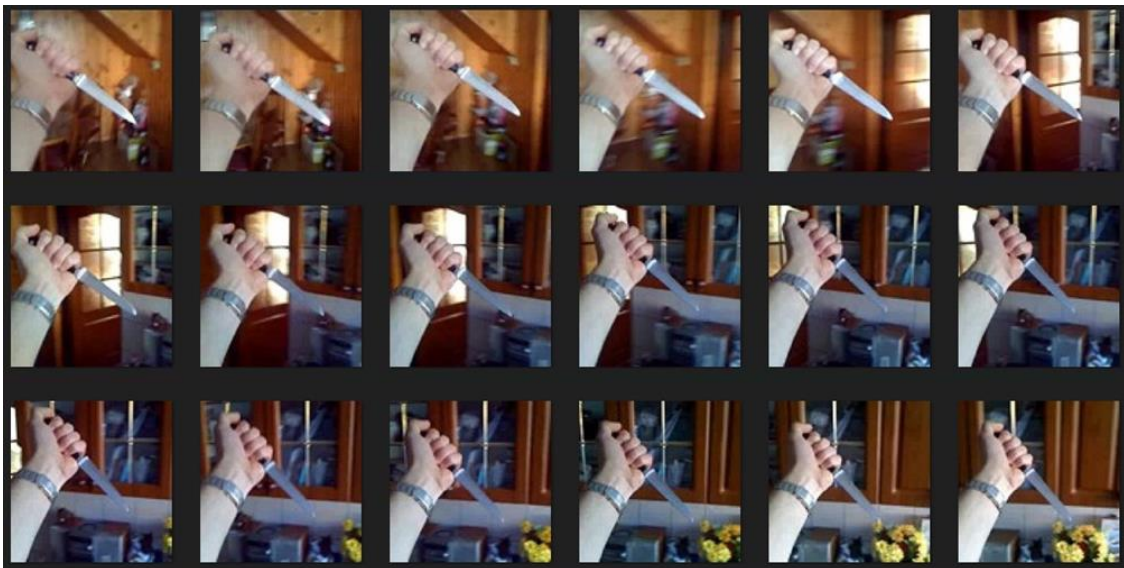


Рисунок 3.4 — Приклади розширення датасету: дзеркальне відображення, масштабування, ротація та зміна освітлення

Для того щоб навчання нейронної мережі моделі YoloV7 було ефективним, кожне зображення потребувало точних анотацій. Було використано спеціалізовані інструменти, LabelImg [96], для ручного маркування зображень, де кожен ніж був обведений у вигляді рамки.

Створення координат об'єкта. Для кожного зображення були визначені координати рамки (рис.3.5), що описують місцезнаходження ножа. Ці координати зберігались у форматі, сумісному з YoloV7, де вказується клас об'єкта (ніж) та позиція рамки.



Рисунок 3.5 — Текстова анотація для зображення у форматі YoloV7

Розмітка різних станів об’єкта. Враховувались випадки, коли ніж частково закривається іншими предметами чи перебуває в руках людини, щоб модель могла розпізнавати об’єкт навіть при частковому перекритті.

Зображення та анотації були поділені на три набори: тренувальний (70%), валідаційний (20%) та тестовий (10%) (рис.3.6). Це дало змогу належно налаштувати модель, оцінити її продуктивність і запобігти перенавчанню.

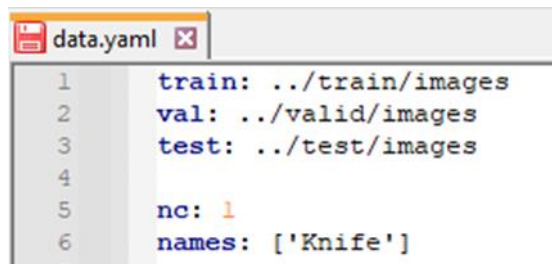


Рисунок 3.6 — Файл data.yaml для налаштування датасетів YoloV7

3.4 Адаптація програмного забезпечення для розпізнавання зброї

На наступному етапі практичного експерименту проведено процес навчання нейронної мережі YoloV7, який є критичним для забезпечення високої точності розпізнавання об’єктів. Навчання моделі проходило з використанням створеного датасету, що складався з 2500 зображень ножів, підготовлених та анотованих для оптимального навчання.

3.4.1 Обмеження класичної моделі YOLO V7

Класична версія моделі YOLOv7 була розроблена для загального виявлення об’єктів і демонструє високу продуктивність у розпізнаванні загальних категорій, таких як автомобілі, люди, тварини тощо [87]. Проте вона

стикається з певними обмеженнями при виявленні специфічних об'єктів, таких як зброя (зокрема ножі та інші небезпечні предмети), особливо в умовах низької видимості або часткового перекриття об'єктів [97].

Недостатня специфічність. Класична YOLOv7 не завжди ефективно розпізнає об'єкти зі специфічною формою або деталями, необхідними для точного ідентифікування зброї. Наприклад, ніж може виглядати подібно до інших предметів, і стандартні нейронні мережі можуть плутати його з нешкідливими об'єктами [98]. Приклад помилкового детектування класичної YoloV7 на рисунку 3.7, де результат показав що в руках у зловмисника замість холодної зброї мобільний телефон.



Рисунок 3.7 — Помилкове детектування класичної YoloV7

Виклики в умовах низької видимості. Класична модель не адаптована для роботи з зображеннями, отриманими за допомогою інфрачервоних камер, і може демонструвати значне зниження точності в умовах поганої освітленості або при наявності перешкод [99]. Це робить необхідним проведення додаткового навчання і покращення архітектури для адаптації до специфічних сценаріїв виявлення зброї.

3.4.2 Процес навчання YOLO V7

Навчання моделі YOLOV7 було здійснено з використанням ретельно підготовленого середовища та налаштувань для досягнення максимальної ефективності. Першим кроком було забезпечення коректної організації датасету, який включав 2500 зображень з анотаціями у форматі Yolo (п.3.4). Усі зображення були розподілені на тренувальний, валідаційний та тестовий набори (рис.3.6) для забезпечення ефективного процесу навчання.

Для запуску навчання було налаштовано відповідний скрипт, який забезпечував передачу всіх параметрів у модель. Код навчання моделі YOLOV7 наведено в додатку Б, пункт 2. Було використано файл конфігурації data.yaml, де були вказані шляхи до тренувального та валідаційного наборів зображень, кількість класів і назви класів (рис.3.6). Модель була завантажена з попередньо навченими вагами для пришвидшення збіжності та кращої узагальнюваності.

Навчання було запущене з такими ключовими параметрами:

- Використані ваги: Початкові ваги yolov7.pt, що забезпечили кращу початкову точку для навчання.
- Файл конфігурації моделі: Вказано файл models/yolov7.yaml, який містив специфікації архітектури нейронної мережі.
- Кількість епох: Навчання проводилося протягом 50 епох, щоб модель мала достатньо часу для збіжності та навчання на всьому датасеті.
- Розмір пакету: Було вибрано розмір пакету, оптимізований для доступних апаратних ресурсів.
- Розмір зображень: Усі зображення були масштабовані до розміру 640x640 пікселів для забезпечення консистентності.

Команда для запуску навчання виглядала наступним чином:

```
python train.py --weights yolov7.pt --cfg
models/yolov7.yaml --data data.yaml --epochs 50 --batch-
size 16 --img-size 640
```

- --weights: Попередньо навчені ваги, які прискорили навчання.

- `--cfg`: Шлях до файлу конфігурації моделі YoloV7.
- `--data`: Шлях до файлу `data.yaml` із зазначенням датасетів та класів.
- `--epochs`: Кількість епох для навчання.
- `--batch-size`: Розмір пакету, оптимізований для апаратного забезпечення.
- `--img-size`: Розмір зображень, який був обраний для тренування.

Під час запуску процес навчання моделі контролювався через виведення в консоль, де можна було спостерігати за значеннями функції втрат та зміною точності моделі (рис.3.8).

```

Anaconda Prompt - python train.py --img 640 --batch 4 --epochs 50 --data C:/Users/mrper/Desktop/Practic/yolov7/data.yaml --weights C:/Users/mrper/Desktop/Practic/yolov7/weights.pt --device cpu
C:\Users\mrper\Desktop\Practic\yolov7\utils\datasets.py:392: FutureWarning: You are using 'torch.load' with 'weights_only=False' (the current default value), which uses the default pickle module implicitly. It is possible to construct malicious pickle data which will execute arbitrary code during unpickling (See https://github.com/pytorch/pytorch/blob/main/SECURITY.md#untrusted-models for more details). In a future release, the default value for 'weights_only' will be flipped to 'True'. This limits the functions that could be executed during unpickling. Arbitrary objects will no longer be allowed to be loaded via this mode unless they are explicitly allowedlist by the user via 'torch.serialization.add_safe_globals'. We recommend you start setting 'weights_only=True' for any use case where you don't have full control of the loaded file. Please open an issue on GitHub for any issues related to this experimental feature.
Model Summary: 487 layers, 37194710 parameters, 37194710 gradients, 185.1 GiB
Transferred 554/560 items from C:/Users/mrper/Desktop/Practic/yolov7/yolov7.pt
Scaled weight_decay = 0.0005
optimizer groups: 95 bias, 95 conv.weight, 92 other
C:\Users\mrper\Desktop\Practic\yolov7\utils\datasets.py:392: FutureWarning: You are using 'torch.load' with 'weights_only=False' (the current default value), which uses the default pickle module implicitly. It is possible to construct malicious pickle data which will execute arbitrary code during unpickling (See https://github.com/pytorch/pytorch/blob/main/SECURITY.md#untrusted-models for more details). In a future release, the default value for 'weights_only' will be flipped to 'True'. This limits the functions that could be executed during unpickling. Arbitrary objects will no longer be allowed to be loaded via this mode unless they are explicitly allowedlist by the user via 'torch.serialization.add_safe_globals'. We recommend you start setting 'weights_only=True' for any use case where you don't have full control of the loaded file. Please open an issue on GitHub for any issues related to this experimental feature.
cache, exists = torch.load(cache_path), True # load
train: Scanning 'C:/Users/mrper/Desktop/Practic/yolov7/Model/train/labels.cache' images and labels... 2514 found, 0 missing, 0 empty, 0 corrupted: 100% | 2514/2514 [00:00<, 71it/s]
val: Scanning 'C:/Users/mrper/Desktop/Practic/yolov7/Model/valid/labels.cache' images and labels... 6675 found, 0 missing, 1 empty, 0 corrupted: 100% | 6675/6675 [00:00<, 71it/s]
autoanchor: Analyzing anchors... anchors/target = 4.41, Best Possible Recall (BPR) = 0.9992
train.py:299: FutureWarning: 'torch.cuda.amp.GradScaler(args...)' is deprecated. Please use 'torch.amp.GradScaler('cuda', args...)' instead.
scaler = amp.GradScaler(enabled=cuda)
image sizes 640 train, 640 test
using 4 dataloader workers
logging results to runs/train/exp5
Starting training for 50 epochs...

  epoch    gpu_mem    box    obj    cls    total    labels    img_size
  -----
0/49      0G      0.07987  0.01959  0  0.09946  6      640: 131E | 0/629 [00:00<, 71it/s]
with amp.autocast(enabled=cuda):
0/49      0G      0.07987  0.01959  0  0.09946  6      640: 131E | 4/629 [00:00<1:45:27, 10.12s/it]

```

Рисунок 3.8 — Консольний вивід під час навчання моделі YoloV7

Проміжні ваги зберігалися автоматично після завершення кожної епохи, що дозволяло оцінити прогрес навчання та забезпечити збереження результатів у разі переривання (рис.3.9).

runs > train > exp5 > weights		
Тип	Ім'я	Розмір
Файл PT	init.pt	145 717 КБ

Рисунок 3.9 — Збережені проміжні ваги моделі YoloV7 після першої епохи

Для навчання моделі проведено 50 циклів (епох). Це означає, що весь датасет був кілька разів переданий через нейронну мережу, що дозволило моделі поступово покращувати свої характеристики та навчитися більш точно ідентифікувати ножі в різних умовах. Кількість епох була обрана на основі

балансування між точністю та часом обчислень, оскільки збільшення кількості епох може призводити до негативного явища як перенавчання мережі моделі.

Після завершення 50 епох точність моделі досягла 90,1%. Цей показник означає, що модель успішно розпізнає ножі у 90,1% випадків з правильною локалізацією та класифікацією об'єкта. Для ілюстрації змін у функції втрат та точності навчання використані таблиці та графіки (таблиця 3.3).

Таблиця 3.3 — Результати втрат та точності по епохах

Епоха	Функція втрат (Loss)	Точність розпізнавання (%) <i>mAP</i>	Час обчислень (хвилини)
1	1,4321	65,2	45,2
5	0,9856	72,4	44,8
10	0,7813	80,1	44,5
15	0,6012	85,3	44,2
20	0,5124	87,9	43,9
25	0,4528	88,6	43,6
30	0,3927	89,3	43,3
35	0,3548	89,8	43,1
40	0,3185	90,0	42,8
45	0,3027	90,1	42,5
50	0,2893	90,1	42,2

Графічне представлення зміни точності до кількості пройдених епох (циклів проходження мережі на навчальному наборі зображень) на рисунку 3.10.

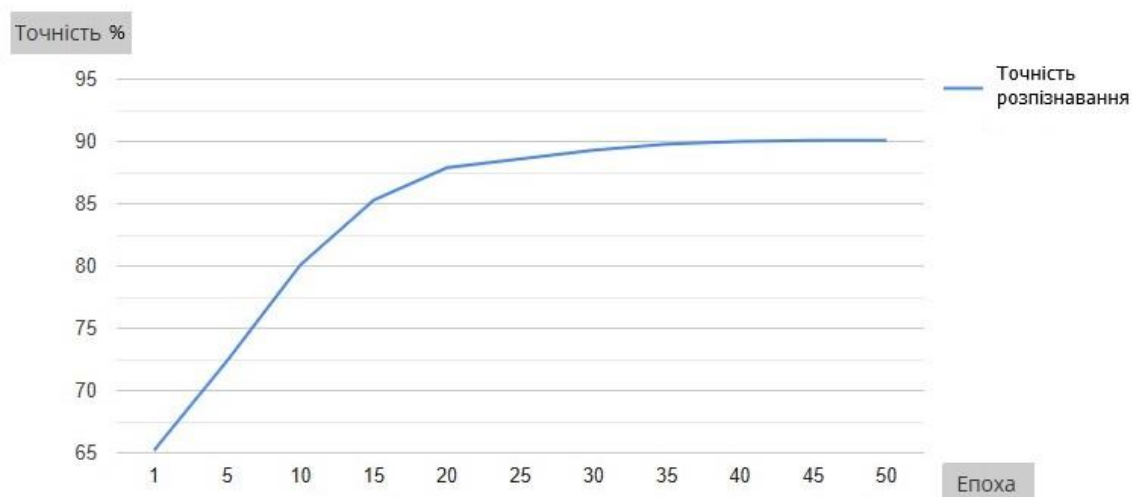


Рисунок 3.10 — Зміна точності до кількості пройдених епох

3.5 Підключення обробки відеопотоку в реальному часі з використанням моделі YOLOV7

Для забезпечення ефективного розпізнавання ножів у реальному часі була реалізована система, яка обробляє відеопотік з камери системи безпеки, яка описана в п.3.1 роботи за допомогою донавченої моделі YOLOV7. Програмний код для обробки відеопотоку в реальному часі наведено в додатку Б, пункт 4. Такий підхід дозволяє виявляти об'єкти безпосередньо під час відеозйомки, що є важливим для застосування в системах відеоспостереження і безпеки.

Обробка відеопотоку в реальному часі складається з декількох ключових компонентів:

Підключення до камери. Використовується камера з підтримкою передавання відео по IP-протоколу, яка передає відеопотік на NVR-сервер (рис.3.1). Захоплення кадрів виконується за допомогою бібліотеки OpenCV (cv2), яка дозволяє ефективно працювати з відеоданими у реальному часі. Відеопотік передавався за допомогою протоколу RTSP, а бібліотека OpenCV використовувалася для захоплення кадрів через метод `cv2.VideoCapture("rtsp://[IP-адреса]")`.

Завантаження та використання моделі YoloV7. Попередньо навчена модель YoloV7 завантажується у середовище PyTorch та переходить у режим

оцінки (evaluation mode). Це дозволяє моделі обробляти зображення швидко і з мінімальними затримками, забезпечуючи високу точність виявлення об'єктів.

Обробка кадрів з відео потоку. Відеопотік захоплюється кадр за кадром, і кожне зображення перетворюється у формат, сумісний з моделлю. Потім кадри передаються через YoloV7, яка виконує розпізнавання об'єктів і повертає координати рамок навколо знайдених ножів. На основі цих даних модифіковані зображення відображаються з накладеними рамками та мітками впевненості. Обробка відеопотоку здійснювалася зі швидкістю 25 кадрів в секунду (FPS), що забезпечувало оперативне розпізнавання об'єктів у реальному часі.

Відображення результатів у реальному часі. Оброблені кадри з виявленими об'єктами відображаються на екрані NVR-клієнта, що дозволяє стежити за розпізнаванням у режимі реального часу. Якщо впевненість моделі у виявленні об'єкта перевищує заданий поріг, на зображенні з'являється рамка навколо ножа та відповідний підпис, наприклад, "Knife: 80%" у нижньому лівому куті.

Реалізована методика дозволяє використовувати нейронну мережу YoloV7 для швидкого розпізнавання ножів, що може допомогти запобігти потенційним загрозам і забезпечити належний рівень безпеки в охоронних системах.

3.6 Логіка роботи системи обробки відеопотоку в електронній системі безпеки

Система обробки відеопотоку в реальному часі з використанням моделі YOLOv7 базується на оптимальному розподілі обчислювальних задач між ключовими компонентами: IP-камерою, процесором і відеокарти комп'ютерної системи NVR-клієнта, дисплеєм (монітором) комп'ютерної системи і сховищем даних (NVR-сервером).

Захоплення відеопотоку. Джерелом відеопотоку є IP-камера Balter моделі IP-MT1210GR, яка передає відеодані через Ethernet-з'єднання на процесор. Завдяки високій роздільній здатності Full HD (1080p) та інфрачервоному підсвічуванню камера забезпечує якісне зображення навіть в умовах низької освітленості. Основними завданнями цього компонента є забезпечення безперервного захоплення кадрів і передача даних для обробки.

Попередня обробка відеопотоку. Процесор AMD Ryzen 5 5600X комп'ютерної системи NVR-клієнта приймає відеопотік із камери для початкової обробки. Він виконує такі функції:

- захоплення кадрів за допомогою бібліотеки OpenCV;
- перетворення формату зображень і масштабування пікселів для подальшого використання нейронною мережею;
- передача підготовлених кадрів на відеокарту для глибокої обробки.

На цьому етапі процесор також координує взаємодію між компонентами системи.

Глибока обробка даних. Відеокарта NVIDIA GeForce RTX 3060 комп'ютерної системи NVR-клієнта є основним обчислювальним модулем для виконання задач глибокого навчання. Її завдання включають:

- використання попередньо навченої моделі YOLOv7 для виявлення об'єктів на кадрах відеопотоку;
- швидке та точне визначення координат об'єктів і рівнів упевненості в результатах;
- повернення результатів обробки до системи.

Відображення результатів. Оброблені дані передаються на дисплей (монітор) для візуалізації. На екрані відображаються кадри відеопотоку з накладеними рамками, які виділяють виявлені об'єкти, та підписами, наприклад: "Knife: 85%". Відображення результатів у реальному часі дозволяє оперативно реагувати на ситуацію.

Збереження даних. Для архівування результатів обробки використовується SSD-накопичувач Samsung 970 EVO Plus, який підключено до NVR-сервера.

Таким чином, запропонована логіка роботи системи дозволяє реалізувати ефективну обробку відеопотоку у реальному часі, забезпечуючи високу продуктивність і точність для систем відеоспостереження та безпеки (рис.3.11).

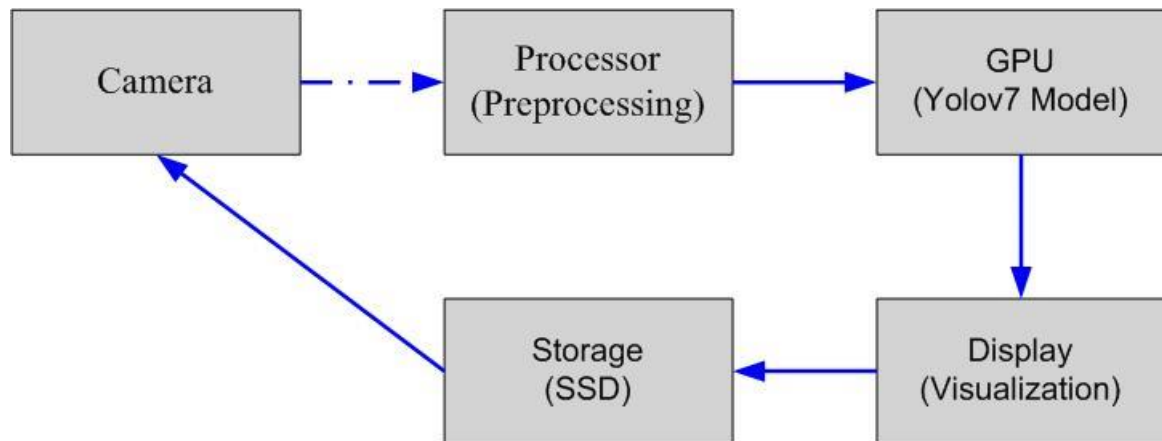


Рисунок 3.11 — Спрощена структура системи обробки відео потоку

3.7 Нові пропозиції в архітектурі моделі YOLO v7

В рамках дослідження для розпізнавання об'єктів холодної зброї, особливо для випадку зменшеної чіткості та при відсутності освітлення (імітація несприятливих умов) середовища, використаємо прийом підключення капсульної мережі (капсульного шару) всередині архітектури моделі YOLO v7 задля вирішення двох проблем – фіксація об'єктів, які частково перекриваються або змінюють свою форму і підвищення точності розпізнавання об'єктів холодної зброї, для випадку зменшеної чіткості і при наявності або відсутності освітлення.

Сучасні дослідження у сфері детекції об'єктів зосереджені переважно на покращенні загальної точності моделей YOLO для видимого спектру зображень [87]. Водночас, застосування капсульних мереж у системах обробки інфрачервоних зображень залишається недостатньо дослідженим, а існуючі

підходи здебільшого орієнтовані на стандартні згорткові нейронні мережі (CNN) без урахування просторових взаємозв'язків між частинами об'єкта [102]. Хоча капсульні мережі були досліджені для задач розпізнавання об'єктів [103], жодне з відомих рішень не використовує капсульні мережі всередині YOLOv7 для підвищення точності детекції холодної зброї в умовах низької освітленості або її відсутності. Таким чином, представлене в цьому дослідженні рішення є інноваційним, оскільки поєднує переваги YOLOv7 у швидкості обробки з можливістю капсульних мереж покращувати розпізнавання об'єктів складної форми, що дозволяє значно підвищити ефективність аналізу безпечності середовища [100].

В основі Capsule Networks лежить концепція капсули, яка є набором нейронів, що представляє вихідний вектор. На відміну від традиційних згорткових нейронних мереж (CNN), де нейрони є скалярними вихідними одиницями, капсули видають вектор із детальною інформацією про об'єкт [102]. Це дозволяє капсульним мережам зберігати не лише факт наявності певного об'єкта, але й його позу (положення, орієнтацію, масштаб), що значно покращує здатність мережі до узагальнення та виявлення об'єктів у складних умовах [104]. Ключовим механізмом у Capsule Networks є routing-by-agreement, який дозволяє капсулам нижчого рівня надсилати інформацію до капсул вищого рівня на основі їхнього взаємного узгодження [105]. Завдяки цьому підходу капсульні мережі можуть ефективніше працювати із зображеннями, де об'єкти змінюють форму, положення або орієнтацію, що є важливим для задач детекції в реальному світі. Структура капсульно нейронної мережі наведена на рисунку 3.12.

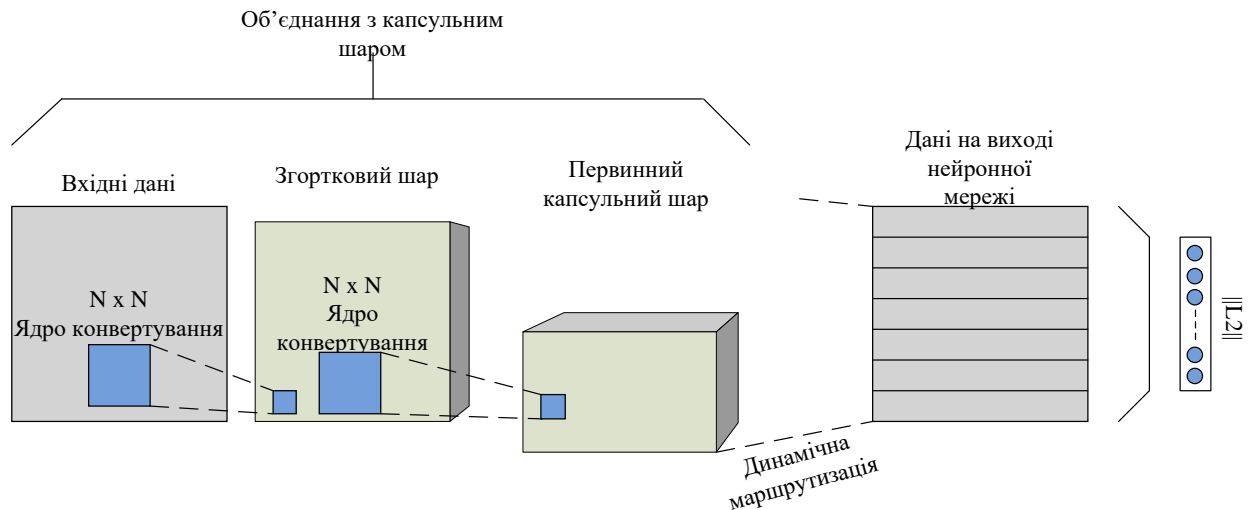


Рисунок 3.12 – Спрощена структурна схема модифікованої моделі [106]

Капсульний шар підключимо після модуля neck в моделі YOLO v7, щоб забезпечити покращене збереження взаємозв'язків між частинами об'єкта. Додатково, прийнято рішення щодо переходу від традиційного скалярного представлення ознак в моделі YOLO v7 до векторного, що дозволяє краще моделювати складні об'єкти.

Перевіримо чи дійсно інтеграція капсульних мереж (Capsule Networks) до архітектури YOLOv7 дозволяє підвищити точність розпізнавання об'єктів із збереженням просторових залежностей. Відомо, що капсульні мережі, на відміну від класичних згорткових нейронних мереж, використовують векторні репрезентації для кодування просторових взаємозв'язків між частинами об'єкта [107]. Додамо після модуля neck архітектури моделі капсульний шар. Крім цього, запропоновано додати наступні зміни в елементи моделі YOLO v7:

- Backbone (залишається без змін): використання CSPDarkNet для витягнення ознак.
- Neck (модифікація): додавання капсульного шару після існуючих convolutional-блоків нейронної мережі моделі для збереження просторових взаємозв'язків.
- Head (адаптація): модифікація класифікаційного модуля для обробки капсульних ознак.

Модифікація для модуля neck моделі YOLO v7 матиме вигляд:

neck:

- [Conv, 256, 1, 1]
- [CapsuleLayer, 8, 256, 32]
- [Conv, 128, 3, 1]

Використання одного капсульного шару в модифікованій архітектурі YOLOv7 обумовлено балансом між точністю, швидкістю обробки та стабільністю навчання.

Капсульний шар можна реалізувати наступним чином:

```
class CapsuleLayer(nn.Module):
    def __init__(self, num_capsules, num_routes, in_dim,
out_dim):
        super(CapsuleLayer, self).__init__()
        self.num_capsules = num_capsules
        self.num_routes = num_routes
        self.W = nn.Parameter(torch.randn(num_capsules,
num_routes, in_dim, out_dim))

    def forward(self, x):
        x = x.unsqueeze(1).unsqueeze(4)
        u_hat = torch.matmul(self.W, x).squeeze(4)
        c = F.softmax(torch.randn(*u_hat.size()), dim=1)
        s = (c * u_hat).sum(dim=1, keepdim=True)
        v = self.squash(s)
        return v.squeeze(1)

    def squash(self, s, epsilon=1e-7):
        norm = torch.norm(s, dim=-1, keepdim=True)
        return (s / (norm + epsilon)) * (norm**2 / (1 +
norm**2))
```


Для перевірки ефективності запропонованих нових змін в архітектуру моделі YOLO v7 проведемо перевірку точності роботи модифікованої моделі. Для цього завантажимо зображення, яке показано на рисунку 3.13. Зауважимо, що на зображенні характерний туман і віддалено біля пішохідного переходу є силуети людей, які перетинаються з контурами.



Рисунок 3.13 – Вхідне тестове зображення

При використанні базової версії моделі YOLO v7 (додаток Б) отримано наступні результати, які показано на рисунку 3.14.



Рисунок 3.14 – Результат при роботі базової версії моделі

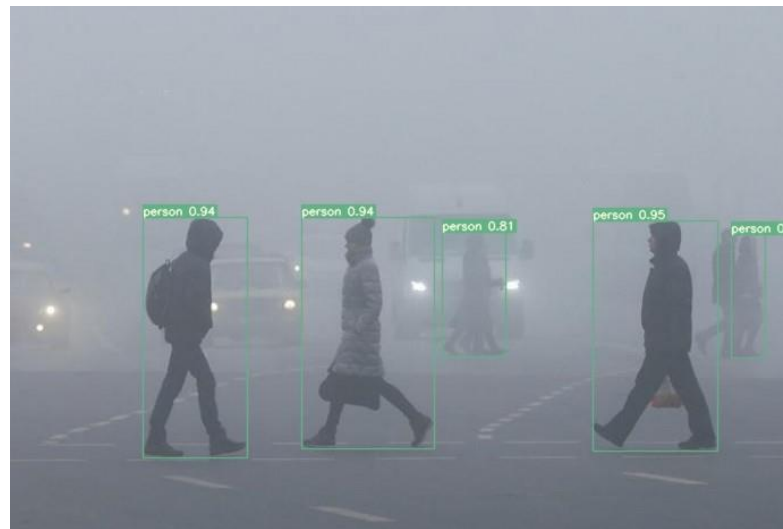
З аналізу отриманих результатів (рис.3.15) видно, що не всі об'єкти були виявлені.

```
4 persons, Done. (698.1ms) Inference, (2.0ms) NMS
The image with the result is saved in: runs\detect\exp2\2.jpg
Done. (0.726s)

(yolov7) C:\Users\mrper\Desktop\Practic\yolov7>
```

Рисунок 3.15 – Результат роботи моделі

При підключенні капсульного шару та запровадження описаних вище налаштувань отримали результат який показано на рисунку 3.16.



```
5 persons, Done. (559.5ms) Inference, (2.0ms) NMS
The image with the result is saved in: runs\detect\exp4\2.jpg
Done. (0.575s)
```

Рисунок 3.16 – Результат при роботі зміненої моделі

Тобто, при порівнянні результатів, виявилось що модель зі зміненою архітектурою працює більш точно, адже вона виявила більше об'єктів. Крім цього, отримано, що модифікована модель працює швидше. Так, для базової версії загальний час роботи складає 0,726 сек, тоді як при змінній архітектурі він виявився 0,575 сек. У підсумку, запропоновані зміни в архітектурі моделі YOLO v7 дозволили отримати кращі показники при розпізнаванні об'єктів в складних умовах зйомки і крім цього, час обробки зображення виявився меншим.

3.8 Перевірка навченої моделі на предмет можливості детектування зброї при наявності освітлення

Для перевірки навченої моделі YoloV7 було проведено серію експериментів, спрямованих на оцінку її здатності розпізнавати зброю (у нашому випадку ножі) в умовах різної відстані та при частковому перекритті об'єкта. Код тестування моделі у різних умовах описано в додатку Б, пункт 3. Метою цього експерименту було визначити, наскільки ефективно модель виявляє об'єкт у реальних сценаріях, які можуть зустрічатися у системах відеоспостереження.

Для зручності у подальшому, в рамках практичного експерименту, в основі алгоритму обраховується метрика (2.3) в перерахунку на відсотки (результат просто множиться на 100%).

Експеримент проводився з використанням 5 серій зображень, кожна з яких представляла ніж у руках людини в різних умовах. Було протестовано два основні сценарії:

- Повне відображення об'єкта: ніж повністю знаходився в полі зору камери, без будь-яких перешкод чи перекриттів.
- Часткове перекриття об'єкта: ніж частково закривався рукою або іншими предметами, що створювало складні умови для розпізнавання.

Зображення були зроблені на різних відстанях від камери, що дозволило оцінити ефективність моделі YoloV7 при зміні віддаленості від об'єкта. Зокрема, були використані такі відстані:

- 1 метр: найближча відстань, при якій ніж добре видно.
- 2 метри: помірна відстань, яка досить часто зустрічається в системах відеоспостереження.
- 3 метри: середня відстань, де починають з'являтися виклики у розпізнаванні.
- 4 метри: відстань, при якій модель може мати проблеми з виявленням через зниження чіткості.

- 5 метрів: найвіддаленіша точка, що дозволяє оцінити, як модель справляється з розпізнаванням на великій відстані.

Модель YoloV7 показала різний рівень ефективності в залежності від відстані до об'єкта та ступеня його перекриття. Далі наведено опис отриманих результатів:

a. відстань 1 метр до камери.

Повне відображення: точність розпізнавання склала 87%, що є високим показником. Модель легко визначила об'єкт та класифікувала його як ніж.

Часткове перекриття: точність трохи знизилася до 79%, але модель все ще змогла ідентифікувати об'єкт завдяки чіткому контуру та текстурі.

b. відстань 2 метри.

Повне відображення: модель зберегла високу точність розпізнавання на рівні 83%, успішно визначивши ніж.

Часткове перекриття: точність знизилася до 75%, що вказує на деякі труднощі у визначенні об'єкта при перекритті.

c. відстань 3 метри.

Повне відображення: точність знизилася до 71%, модель все ще успішно виявляла ніж, але з дещо меншою впевненістю.

Часткове перекриття: рівень розпізнавання впав до 69%, оскільки ніж став менш виразним на фоні.

d. відстань 4 метри.

Повне відображення: точність далі зменшилася до 59%, модель мала складності з визначенням меж об'єкта.

Часткове перекриття: точність значно впала до 48%, оскільки ніж ставав складним для розпізнавання.

e. відстань 5 метрів.

Повне відображення: точність виявлення об'єкта склала 33 %, що свідчить про значне зниження ефективності моделі.

Часткове перекриття: об'єкт виявлений але точність складає лише 18%, що підкреслює вже обмеження моделі.

Отримані результати показують, що модель YoloV7 добре працює на коротких відстанях, особливо коли ніж повністю видно у кадрі. Однак при збільшенні відстані та частковому перекритті об'єкта точність розпізнавання знижується. Випадок з відстанню у 5 метрів і частковим перекриттям продемонстрував, що модель ідентифікує ніж, що може бути недостатнім.

Результати розпізнавання ножа моделлю YoloV7 при різних відстанях та умовах перекриття без капсульного шару, та з додаванням цього елемента представлені в таблиці 3.4.

Дані з таблиці 3.4 наочно демонструють, як змінюється точність моделі залежно від відстані до об'єкта та ступеня перекриття. Тобто отримано зниження точності з віддаленням від об'єкта, особливо в умовах часткового перекриття.

Таблиця 3.4 – Результати розпізнавання ножа моделлю YoloV7 при різних відстанях та умовах перекриття

Умови, відстань (м)	Точність розпізнавання (%) без капсульного шару	Точність розпізнавання (%) з капсульним шаром
Повне відображення, 1 м	87%	89%
Часткове перекриття, 1 м	79%	84%
Повне відображення, 2 м	83%	85%
Часткове перекриття, 2 м	75%	80%
Повне відображення, 3 м	71%	74%
Часткове перекриття, 3 м	69%	73%

Продовження таблиці 3.4 – Результати розпізнавання ножа моделлю YoloV7 при різних відстанях та умовах перекриття

Умови, відстань (м)	Точність розпізнавання (%) без капсульного шару	Точність розпізнавання (%) з капсульним шаром
Повне відображення, 4 м	59%	63%
Часткове перекриття, 4 м	48%	56%
Повне відображення, 5 м	33%	38%
Часткове перекриття, 5 м	18%	25%

З аналізу отриманих даних таблиці 3.4 можна зробити висновок, що додавання одного капсульного шару в архітектуру моделі YOLOv7 при наявності освітлення та нормальних умов фіксації призвело до підвищення середньої оцінки точності приблизно на 2-5% (рис.3.17).

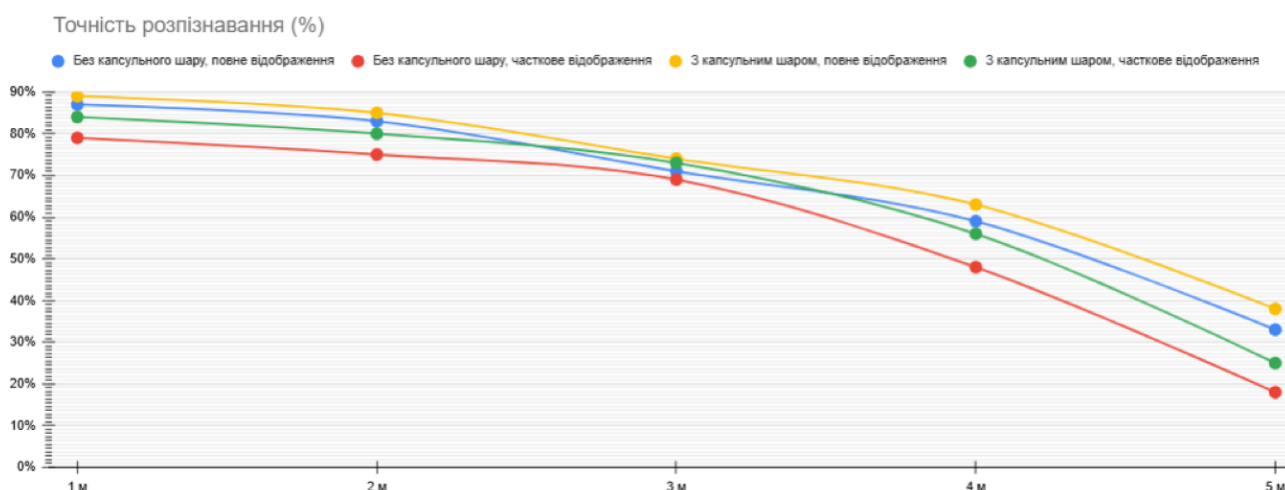


Рисунок 3.17 — Зміна точності розпізнавання (%) в умовах наявності освітлення при різних відстанях та умовах перекриття у двох варіантах (з/без) капсульного шару

Зображення з результатами розпізнавання ножа при наявності освітлення моделлю YoloV7 при різних відстанях та умовах перекриття наведені в додатку Б, пункт 5. Перевіримо далі чи будуть отриманні кращі результати розпізнавання спеціального об'єкту зброї для випадку, коли зменшена видимість зображення на екрані камери але освітлення в приміщенні не відключено.

3.9 Перевірка моделі в умовах зменшеної чіткості при наявності освітлення (з імітацією несприятливих умов фіксації)

Наступний етап експерименту щодо перевірки модифікованої моделі YoloV7 проводився в умовах, коли видимість була значно погіршена. Було зменшено прозорість зображень ще більше, що імітувало екстремально погані умови, такі як сильний туман, або просто штучне задимлення зловмисником в приміщенні. Мета цього експерименту полягала у визначенні, як модель реагує на суттєве погіршення якості зображення.

Зображення були зроблені на відстанях 1-5 метр, причому на відстані 4 м усереднена точність була досить низькою. Кожне зображення зазнало значного зниження видимості через надмірне зменшення прозорості, що створило умови, максимально наближені до реальних ситуацій з поганою видимістю.

a. відстань 1 метр.

Повне відображення: модель виявила ніж із точністю 53%, що свідчить про суттєве зниження впевненості.

Часткове перекриття: точність впала до 37%, ідентифікація була майже випадковою.

b. відстань 2 метри.

Повне відображення: точність становила лише 32%, модель значно втрачала можливість розпізнавати об'єкт у таких умовах.

Часткове перекриття: об'єкт був ідентифікований із точністю лише 28%, що майже не перевищує випадкове вгадування.

с. відстань 3 метри.

Повне відображення: об'єкт ідентифікований з точністю 20% при повному відображенні та 16% при частковому перекритті.

Що стосується відстаней 4 м та 5 м, хоча об'єкт і був визначений в умовах зменшеної чіткості при наявності освітлення, проте отримані дані точності є дуже малими (табл.3.5).

Результати розпізнавання ножа моделлю YoloV7 в умовах зменшеної прозорості при наявності освітлення (з імітацією поганих умов) відображені в таблиці 3.5. Для даного етапу експерименту, так само розглянуто дві архітектури нейронної мережі при підключенні капсульного шару, та за умови його відсутності.

Таблиця 3.5 — Результати розпізнавання ножа моделлю YoloV7 в умовах зменшеної чіткості та наявності освітлення при різних відстанях та умовах перекриття

Умови, відстань (м)	Точність розпізнавання (%) без капсульного шару	Точність розпізнавання (%) з капсульним шаром
Повне відображення, 1 м	53%	58%
Часткове перекриття, 1 м	37%	45%
Повне відображення, 2 м	32%	38%
Часткове перекриття, 2 м	28%	33%
Повне відображення, 3 м	20%	26%
Часткове перекриття, 3 м	16%	22%
Повне відображення, 4 м	9%	14%

Продовження таблиці 3.5 – Результати розпізнавання ножа моделлю YoloV7 в умовах зменшеної чіткості та наявності освітлення при різних відстанях та умовах перекриття

Часткове перекриття, 4 м	6%	10%
Повне відображення, 5 м	3%	7%
Часткове перекриття, 5 м	2%	5%

Дані таблиці 3.5 показують, як різке зниження видимості вплинуло на здатність моделі розпізнавати ніж. На відстані 5 метри об'єкт був виявлений з досить низькою точністю розпізнавання, що підкреслює обмеження моделі у складних умовах. Порівнюючи отримані результати за даними таблиці 3.5 можна переконатись у тому, що покращення даних точності за відстанях до 3 м при додаванні капсульного шару в середньому склало 6%. Зображення з результатами розпізнавання ножа при наявності освітлення та імітацією несприятливих умов моделлю YoloV7 при різних відстанях та умовах перекриття наведені в додатку Б, пункт 6.

Цей експеримент показав, що модель YoloV7 виявляє значні труднощі при розпізнаванні об'єктів у надмірно поганих умовах видимості. Зокрема, точність виявлення значно знизилася вже на відстані 2 метра, а на відстані 5 метри розпізнавання практично дорівнює нулю. Це свідчить про необхідність удосконалення моделі, наприклад, використання додаткових технологій, таких як теплові сенсори або комбіновані системи розпізнавання, для забезпечення безпеки в екстремальних умовах.

3.10 Адаптація програмного забезпечення для розпізнавання зброї (ножів) з інфрачервоного зображення

Після успішного навчання моделі YoloV7 на стандартних зображеннях ножів виникла необхідність зробити модель ефективнішою для розпізнавання об'єктів в умовах низької видимості, зокрема на зображеннях, отриманих за допомогою камери нічного бачення. Оскільки системи безпеки часто використовують інфрачервоні камери для моніторингу у темряві, адаптація моделі до таких специфічних умов стала важливим етапом у моєму дослідженні.

Сучасні алгоритми детекції об'єктів, такі як YOLOv7, демонструють високу ефективність при роботі з видимим спектром, але їх застосування для аналізу інфрачервоних (IR) зображень є значно обмеженим [87]. Це зумовлено тим, що традиційні моделі обробки зображень не адаптовані до специфіки інфрачервоного спектру, де відсутня кольорова інформація, а контрастність об'єктів є нижчою [100]. Крім того, на момент проведення цього дослідження відсутні готові рішення на основі YOLOv7 для детекції холодної зброї на інфрачервоних зображеннях, що робить цю задачу особливо актуальною [101].

Камера нічного бачення, яка була використана, створює чорно-білі зображення з високою контрастністю, підсвічуючи об'єкти за допомогою інфрачервоного світла. У темряві ця камера дозволяє чітко бачити об'єкти, однак відсутність кольору та інфрачервоне підсвічування ускладнюють процес автоматичного розпізнавання.

Приклад інфрачервоного зображення, отриманого з камери нічного бачення на рисунку 3.17.



Рисунок 3.18 — Приклад інфрачервоного зображення з камери

Підготовка до донавчання моделі здійснювалась наступним чином. Було зібрано інфрачервоні зображення ножів за допомогою камери нічного бачення. Ці зображення відрізнялися від звичайних тим, що були чорно-білими та мали високу контрастність. Для збільшення різноманітності датасету було додано зображення, зняті з різних ракурсів і в умовах змінного підсвічування (рис.3.18).

Щоб модель стала більш стійкою до варіацій у вигляді об'єктів, було використано аугментацію. Це включало обертання зображень, зміну контрастності та додавання штучного шуму (рис.3.20).

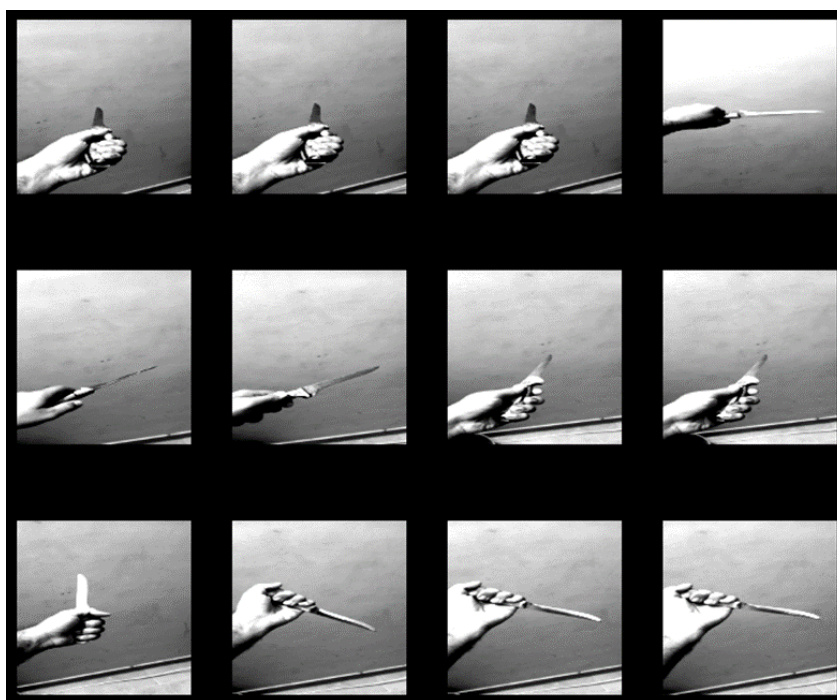


Рисунок 3.19 — Зображення, зняті з різних ракурсів

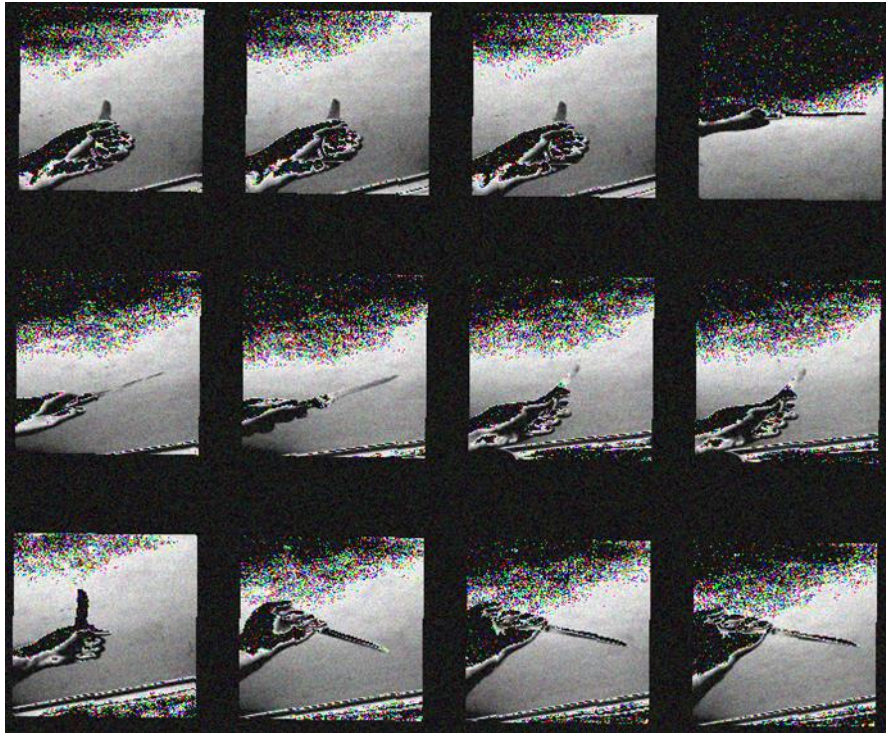


Рисунок 3.20 — Зображення після зміни контрастності та додавання штучного шуму

Після початкового навчання моделі YoloV7 для розпізнавання ножів у стандартних умовах було проведено донавчання цієї ж моделі для адаптації до інфрачервоних зображень. Це дозволило моделі зберегти свої початкові знання про ножі та поліпшити здатність працювати в умовах низької видимості. Команда для запуску донавчання виглядала наступним чином:

```
python train.py --weights my_trained_model.pt --cfg
models/yolov7.yaml --data data.yaml --epochs 30 --batch-
size 16 --img-size 640
```

Процес донавчання тривав 30 епох, протягом яких спостерігалось поступове покращення точності розпізнавання. Модель змогла краще адаптуватися до специфіки інфрачервоних зображень, що підтверджувалося зниженням функції втрат та підвищенням точності класифікації до 80%.

Нижче наведено результати втрат та точності моделі протягом 30 епох (таблиця 3.6).

Таблиця 3.6 — Результати втрат та точності моделі протягом 30 епох донавчання

Епоха	Функція втрат (<i>Loss</i>) (2.1)	Точність розпізнавання (%)
1	1,2345	55,4
5	0,9567	63,2
10	0,7689	70,1
15	0,6532	73,5
20	0,5897	76,3
25	0,5234	78,4
30	0,4789	80,0

Графічне представлення зміни точності моделі протягом 30 епох донавчання відображено на рисунку 3.17.

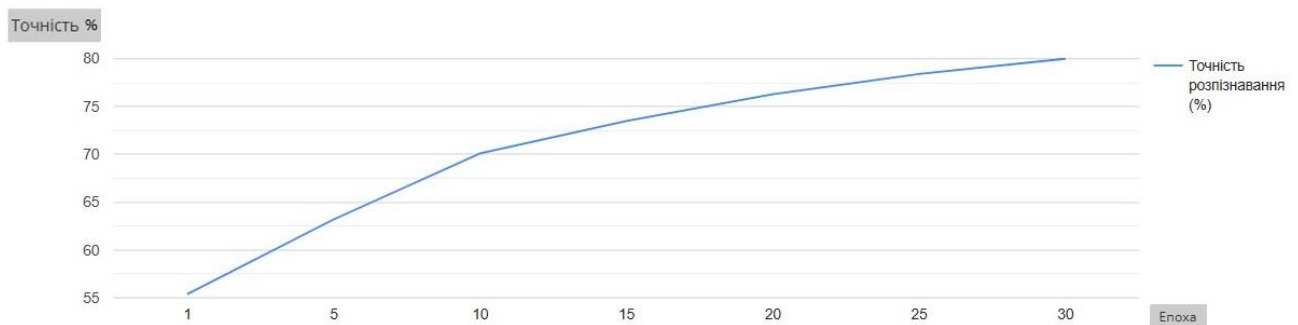


Рисунок 3.21 — Зміна точності моделі протягом 30 епох

Донавчання моделі YoloV7 дозволило значно покращити її продуктивність у розпізнаванні ножів на інфрачервоних зображеннях. Завдяки збереженню початкових знань і адаптації до нових умов, модель тепер здатна розпізнавати об'єкти з точністю 80%, що є суттєвим покращенням для роботи в умовах низької видимості.

3.11 Перевірка навченої моделі на предмет можливості детектування зброї при відсутності освітлення

Для перевірки донавченої моделі YoloV7 було проведено серію тестів у повній темряві з використанням інфрачервоного підсвічування камери нічного бачення. Експеримент мав на меті визначити, наскільки точно модель розпізнає ножі на різних відстанях від камери, включаючи випадки часткового перекриття об'єкта.

Тести проводилися в умовах повної темряви, і для кожної відстані виконувалися два основних сценарії:

- повне відображення об'єкта: ніж повністю знаходився у полі зору камери, без жодних перекриттів.
- часткове перекриття об'єкта: ніж частково закривався рукою або іншими предметами.

Були отримані такі результати експерименту.

a. відстань 1 метр.

Повне відображення: модель з високою точністю 81% розпізнавала ножі навіть за змінних ракурсів.

Часткове перекриття: точність знизилася до 67%, проте модель успішно визначила контури ножа.

b. відстань 2 метри.

Повне відображення: модель зберегла точність на рівні 77%, хоча впевненість у класифікації почала знижуватися.

Часткове перекриття: точність впала до 61%, що вказує на складнощі у визначенні об'єкта при частковому закритті.

c. відстань 3 метри.

Повне відображення: точність моделі впала до 50%, і впевненість у виявленні зменшилася.

Часткове перекриття: точність впала до 22%, що вказує вже на труднощі у визначенні об'єкта при частковому закритті.

d. Відстань 4 метри.

Точність розпізнавання впало до величини 15% при повному відображенні, а при частковому відображенні склало лише 11%.

e. Відстань 5 метрів.

Повне відображення: ніж був розпізнаний на рівні лише 9%, коли лезо блиснуло в полі підсвічування, однак загальна впевненість залишалася дуже низькою.

Часткове перекриття: модель ідентифікувати об'єкт на рівні 6% впевненості.

Донавчена модель YoloV7 продемонструвала високу ефективність лише на коротких відстанях (1-2 метри) при повному відображенні ножа. Проте, з віддаленням об'єкта і при частковому перекритті точність значно знизилася. Відстань понад 4 метри створює серйозні виклики для моделі, особливо в умовах обмеженого інфрачервоного освітлення.

Результати розпізнавання ножа моделлю YoloV7 при різних відстанях та умовах перекриття в умовах відсутності освітлюваності для двох архітектур моделі YOLOv7 – при підключенні капсульного шару та за умови його відсутності (базова модель архітектури) представлені в таблиці 3.7.

Таблиця 3.7 — Результати розпізнавання ножа в умовах відсутності освітлюваності

Умови, відстань (м)	Точність розпізнавання (%) без капсульного шару	Точність розпізнавання (%) з капсульним шаром
Повне відображення, 1 м	81%	85%
Часткове перекриття, 1 м	67%	73%
Повне відображення, 2 м	77%	81%
Часткове перекриття, 2 м	61%	67%

Продовження таблиці 3.7 – Результати розпізнавання ножа в умовах відсутності освітлюваності

Повне відображення, 3 м	48%	56%
Часткове перекриття, 3 м	22%	30%
Повне відображення, 4 м	15%	21%
Часткове перекриття, 4 м	11%	17%
Повне відображення, 5 м	9%	12%
Часткове перекриття, 5 м	6%	9%

Ці результати з таблиці 3.7 підкреслюють необхідність використання додаткових підходів для покращення виявлення об'єктів на великих відстанях і при складних умовах видимості. Зображення з результатами розпізнавання ножа при відсутності освітлюваності моделлю YoloV7 при різних відстанях та умовах перекриття наведені в додатку Б, пункт 7.

3.12 Перевірка моделі в умовах зменшеної чіткості і при відсутності освітлення (імітація несприятливих умов)

Для більш глибокого аналізу було вирішено додатково провести тести в умовах імітації поганої видимості. Зменшення чіткості на 50% мало на меті створити умови, подібні до тих, що виникають при задимленості, тумані або інших перешкодах.

Випробування проводилися аналогічно попередньому експерименту, з об'єктами на відстані від 1 до 5 метрів. Застосовувалися два основні сценарії:

- повне відображення об'єкта: ніж повністю видимий у кадрі;
- часткове перекриття об'єкта: ніж частково закривався, що створювало додаткові труднощі для моделі.

Відстань 1 метр.

- Повне відображення: модель продемонструвала помітне зниження точності до 32%, хоча все ще успішно розпізнавала ніж.
- Часткове перекриття: точність знизилася до 20%, але модель змогла ідентифікувати об'єкт.

Відстань 2 метри.

- Повне відображення: точність розпізнавання впала до 25%, і модель демонструвала деякі помилкові спрацьовування.
- Часткове перекриття: ніж розпізнаний за значенням схожості 12%.

Відстань 3 метри.

- Повне відображення: точність знизилася до 8%,
- Часткове перекриття: точність знизилася до 6%.

Дані для відстаней 4 м та 5 м наведено у таблиці 3.8. Результати показали, що зменшення прозорості на 50% значно ускладнює процес розпізнавання об'єктів. Модель починає втрачати точність вже на відстані 3 метри, а при частковому перекритті ефективність розпізнавання різко падає. Результати розпізнавання ножа моделлю YoloV7 в умовах зменшеної прозорості при відсутності освітлення (імітація поганих умов) відображені в таблиці 3.8.

За даними з таблиці 3.8 виявилось, що на малих відстанях об'єкту від камери (до 3 м) вираш у підключенні додаткового капсульного шару за середніми значеннями точності mAP у відсотках склав приблизно 3-5%. Зображення з результатами розпізнавання ножа при відсутності освітлюваності та імітацією несприятливих умов моделлю YoloV7 при різних відстанях та умовах перекриття наведені в додатку Б, пункт 8.

Таблиця 3.8 — Результати розпізнавання ножа моделлю YoloV7 в умовах зменшеної чіткості та відсутності освітлення при різних відстанях

Умови, відстань (м)	Точність розпізнавання (%) без капсульного шару	Точність розпізнавання (%) з капсульним шаром
Повне відображення, 1 м	32%	38%
Часткове перекриття, 1 м	20%	27%
Повне відображення, 2 м	25%	30%
Часткове перекриття, 2 м	12%	17%
Повне відображення, 3 м	8%	12%
Часткове перекриття, 3 м	6%	9%
Повне відображення, 4 м	4%	7%
Часткове перекриття, 4 м	0%	0%
Повне відображення, 5 м	0%	0%
Часткове перекриття, 5 м	0%	0%

Проведені експерименти в умовах повної темряви та з імітацією поганої видимості дозволили оцінити здатність донавченої моделі YoloV7 розпізнавати зброю, зокрема ножі, у різних сценаріях. Результати показали, що модель демонструє задовільну продуктивність на коротких відстанях (1-2 метри) при

повному відображенні об'єкта, навіть у повній темряві, але її ефективність значно знижується при збільшенні відстані та частковому перекритті об'єкта.

У стандартних умовах нічного бачення модель виявляла об'єкти з високою точністю на коротких дистанціях. Проте на відстані понад 3 метри і при частковому перекритті точність значно падала, що вказує на обмежену здатність моделі до розпізнавання об'єктів на великих відстанях.

Імітація поганих умов видимості через зменшення прозорості на 50% суттєво ускладнила розпізнавання. Модель швидко втрачала ефективність навіть на відстані 3 метри, особливо при частковому закритті об'єкта. Точність розпізнавання об'єктів у таких умовах впала до рівнів, які є недостатньо надійними для використання у критичних системах безпеки.

Значно кращі результати роботи моделі YOLOv7 отримані при додаванні в архітектуру капсульного шару. Зазначимо, що для забезпечення точності експерименту для зміненої моделі не проводилось додаткове навчання мережі і були використані ті самі зображення, які були визначені у п.3.7-3.11 роботи, причому будуть перевірені лише ті випадки, коли стандартна модель YOLO v7 показала незадовільні результати з розпізнавання об'єкту холодної зброї.

Таким чином, запроваджені зміни в структуру моделі YOLO v7 дозволили отримати кращі значення точності з розпізнавання об'єкту холодної зброї.

Таблиця 3.9 — Результати розпізнавання ножа моделлю YoloV7 з капсульним шаром при повному відображенні об'єкта на відстані 3м в різних умовах

Умови, відстань (м)	Точність розпізнавання (%) з капсульним шаром, повне відображення 3м	Зображення
Звичайне освітлення	74%	
Зменшена чіткість при наявності освітлення	26%	
Відсутнє освітлення	56%	

Продовження таблиці 3.9 – Результати розпізнавання ножа моделлю YoloV7 з капсульним шаром при повному відображенні об'єкта на відстані 3м в різних умовах

Зменшена чіткість при відсутності освітленості	12%	
---	-----	--

3.13 Аналіз отриманих результатів практичного експерименту

Знайдено, що за умовами першого етапу практичного експерименту при наявності освітлення на 1 метрі базова модель YOLOv7 досягла точності 87% при повному відображенні ножа, а з додаванням капсульного шару точність модифікованої моделі YOLOv7 покращилася вже до 89%. У разі часткового перекриття об'єкта точність знизилася до 79%, але з використанням капсульного шару вона підвищилася до 84%. На відстані 2 метри від камери електронної системи безпеки (рис.3.1) модель зберегла достатньо високу ефективність – 83% при повному відображенні ножа та 75% при частковому перекритті. Додавання капсульного шару в цьому випадку дозволило підвищити усереднену точність розпізнавання до 85% і 80% відповідно. При збільшенні відстані до 3 метрів спостерігалось подальше зниження точності – 71% для повного відображення та 69% при частковому перекритті, тоді як капсульний шар сприяв незначному зменшенню показників *mAP* до 74% і 73% відповідно. Таким чином, можна стверджувати, що модель ефективно розпізнає ножі на близьких відстанях, особливо за умови повного відображення об'єкта, однак часткове перекриття та збільшення дистанції призводять до поступового зниження точності.

В умовах відсутності освітлення (етап 2 практичного експерименту) модель також продемонструвала високу ефективність на відстанях до 3 метрів, хоча її точність була нижчою, порівняно з тестуванням при освітленні (етап 1). На відстані 1 метр модель досягла точності 81% при повному відображенні та 67% у випадку часткового перекриття. Додавання капсульного шару за цих умов дозволило покращити результати для цієї відстані до 85% і 73% відповідно. На відстані 2 метрів точність виявлення ножа при повному відображенні становила 77%, а при частковому перекритті – 61%, тоді як використання капсульного шару сприяло підвищенню показників при порівнянні до 81% і 67% відповідно. На 3 метрах модель закономірно почала демонструвати значне зниження точності – 50% при повному відображенні та 22% у випадку часткового перекриття, проте капсульний шар дозволив отримати менші величини зниження точності – 56% і 30% відповідно. Це свідчить про те, що модель залишається ефективною для розпізнавання ножів на коротких дистанціях, навіть в умовах відсутності освітлення, але зі збільшенням відстані та при частковому перекритті точність помітно знижується.

Проведені експерименти з перевірки моделі YOLOv7 та її запропонованої модифікації в умовах зменшеної чіткості і при наявності освітлення показали значне зниження ефективності розпізнавання ножа, особливо на відстанях понад 1 метр. При повному відображенні об'єкта на 1 метрі модель продемонструвала точність 53%, а з використанням капсульного шару – 58%. У разі часткового перекриття точність різко впала до 37%, але було отримано точність 45% після додавання капсульного шару за цих умов. Зі збільшенням відстані до 2 метрів, точність при повному відображенні знизилася до 32% (38% із капсульним шаром), а при частковому перекритті – до 28% (33% із капсульним шаром). На 3 метрах модель втратила ще більше точності – 20% при повному відображенні та 16% при частковому перекритті, тоді як капсульний шар забезпечив лише незначне покращення до 26% і 22%

відповідно. Отримані результати демонструють, що зниження чіткості суттєво ускладнює процес розпізнавання ножа, навіть на близьких відстанях.

Перевірка моделі в умовах зменшеної чіткості та відсутності освітлення підтвердила значні труднощі в розпізнаванні об'єктів. На відстані 1 метр при повному відображенні точність розпізнавання впала до 32% без капсульного шару та до 38% із ним, а при частковому перекритті показники склали 20% і 27% відповідно. На 2 метрах точність знизилася ще більше – 25% без капсульного шару та 30% із ним при повному відображенні, а при частковому перекритті лише 12% і 17% відповідно. На 3 метрах модель майже втратила здатність розпізнавати об'єкт: 8% при повному відображенні та 6% при частковому перекритті, з незначним покращенням до 12% і 9% після додавання капсульного шару. Отримані результати свідчать, що в умовах низької видимості модель втрачає точність вже на відстані 2-3 метрів, а часткове перекриття ще більше ускладнює процес ідентифікації. Це підтверджує необхідність додаткових методів покращення розпізнавання, таких як комбіновані системи аналізу або альтернативні сенсори.

Таким чином, на основі отриманих результатів можна стверджувати, що додавання капсульного шару в архітектуру моделі YOLOv7 призвело до збільшення усереднього значення метрики точності у відсотковому відношенні при різних відстанях спеціального об'єкту до камери в середньому на 4-6%. Крім цього, можна переконатись, що зі збільшенням відстані від об'єкту до камери, вже починаючи зі значення 3 м, особливо при імітації несприятливих умов фіксації наявне зменшення значень точності ідентифікації об'єкту.

3.14 Оцінка ефективності модифікованої моделі YOLOv7

Оцінка ефективності моделі на основі метрик дозволяє зрозуміти наскільки ефективно модель функціонує. Тобто, необхідно перевірити чи є розроблена модель з капсульним шаром точною та ефективною. Для вимірювання ефективності моделі використовуються різні метрики [108], серед

яких, вище була використана у дослідженні, метрика mAP , на основі співвідношення (2.3), яка визначає показник впевненості моделі YOLO v7 у тому, що виявлений об'єкт належить до певного класу. Цей показник впевненості дозволяє відкидати прогнозовані значення, які менше за обраний поріг. Тобто, лише ті виявлення з показником впевненості більше за обране порогове значення можна вважати достовірними. Разом з тим, для спеціалізованого об'єкту типу “ніж”, в рамках проведеного дослідження у п.38-3.12 (для випадків, коли поріг впевненості $\geq 50\%$) ця метрика завжди буде дорівнювати 100%, оскільки сам об'єкт є детектованим і параметр $FP = 0$. Враховуючи це, для оцінки точності моделі використаємо іншу оцінку $F1$, яку можна визначити на основі такої формули [108]:

$$F1 = 2 \times \frac{mAP \times R}{mAP + R} \times 100\%, \quad (3.1)$$

де

$$R = \frac{TP}{TP + FN} \times 100\%,$$

де TP – кількість випадків, коли впевненість моделі $\geq 50\%$; FN – кількість випадків, коли модель має впевненість $< 50\%$. В таблиці 3.10 наведено отримані значення метрики (3.1) для проведеного експерименту при визначеному порозі впевненості у 50%.

Таблиця 3.10 — Результати визначення метрики $F1$

Умова проведення	Наявність чи ні капсульного шару в моделі	Параметр R	Метрика $F1$
Звичайна фіксація зображення	-	70%	82%
	+	80%	89%
Освітлення є, але зменшена видимість	-	10%	18%
	+	10%	18%

Продовження таблиці 3.10 — Результати визначення метрики $F1$

Умова проведення	Наявність чи ні капсульного шару в моделі	Параметр R	Метрика $F1$
Немає освітлення	-	50%	67%
	+	50%	67%
Немає освітлення та зменшена видимість	-	0	0
	+	0	0

Аналізуючи отримані значення метрики за даними таблиці 3.10 можна дійти висновку, що проглядається взаємозв'язок з встановленими умовами проведення фіксації зображення. Разом з тим, за умов нормального освітлення в приміщенні, при функціонуванні електронної системи безпеки, отримано, що метрика $F1$ майже на 7% є більшою за відповідне значення, коли капсульний шар в структурі моделі YOLOv7 не застосовується. Отримано підтвердження тези [108], що метрика $F1$ показує збалансовану оцінку продуктивності моделі.

Висновки до розділу 3.

1. Сформовано спеціалізований навчальний набір зображень холодної зброї, які включали різні ракурси, умови освітлення, фони та рівні шуму. Усі зображення були анотовані вручну, із точним визначенням координат об'єктів у форматі роботи моделі.

2. Проведено навчання моделі YoloV7 тривалістю 50 епох із використанням попередньо навчених ваг. В результаті точність розпізнавання ножів на навчанні досягла 90,1%, при цьому час обробки кожного зображення скоротився до 42,2 хвилин для однієї епохи.

3. Проведена перевірка навченої моделі YOLOv7 при різних умовах освітлення на обраних відстанях від 1 до 3 метрів. Розроблено змінену модель YOLO v7 шляхом додавання в архітектуру нейронної мережі моделі одного

капсульного шару. Отримано, що завдяки такому технічному прийому можна вирішити проблеми реєстрації об'єктів, які частково перекриваються або змінюють свою форму і забезпечити підвищення точності розпізнавання об'єктів холодної зброї в умовах зменшеної чіткості зображення, яке отримується з камери системи безпеки при наявності та відсутності освітлення. Проведена оцінка ефективності розробленої моделі на основі обрахунку метрики $F1$.

ВИСНОВКИ

За результатами проведеного дослідження отримано такі результати:

1. Проведено порівняння двох найбільш популярних алгоритмів CAMShift та Optical Flow для розв'язання задачі відстеження об'єктів на обраних динамічних кадрах зображення зі складним фоновим рисунком. Знайдено, що обидва алгоритми вимагають, щоб кадри зображення для аналізу відповідали кольоровій моделі HSV. За заданими параметрами перевірених наборів динамічних зображень знайдено, що для алгоритму CAMShift властива чутливість до змін кольорових характеристик та освітлення. Алгоритм Optical Flow водночас, менш чутливий до зміни кольорових характеристик та шуму, проте є більш складним у налаштуванні і вимагає більше обчислювальних ресурсів. І це пов'язано, як відмічено, з додатковими розрахунками векторів зміщень для кожного найменшого елемента зображення у кожному кадрі.

2. Наведено програмний алгоритм зменшення яскравісного шуму на поверхні зображення шляхом застосування нейронної мережі, яка складається з 40 згорткових шарів та поділена за функціональними властивостями, на три частини. Отримано, що запропонована структура нейронної мережі добре адаптована до етапу навчання. Так, для проведених 600 циклів навчання отримане значення функції втрат для двох наборів тестових зображень не перевищило величини 10^{-3} . Обчислено показники якості оброблення обраного тестового зображення з яскравісним шумом. Так, для значення дисперсії шуму 10, індекс структурної подібності *SSIM* за рахунок використання нейронної мережі був збільшений з 0,9652 до 0,9953. Пікове відношення сигнал/шум для цього ж зображення збільшилось з величини 36,13 дБ до 47,16 дБ. В цілому за рахунок використання нейронної мережі згорткового типу метрика пікового відношення сигнал-шум PSNR після обробки на 4-5 дБ перевищує значення PSNR до проведення обробки.

На основі використаного алгоритму з застосуванням вейвлет-перетворення Хаара запропоновано алгоритм виділення та розпізнавання тексту

з кольорового зображення. Отримано, що алгоритм розпізнавання текстових символів з використанням цього алгоритму виявився стійким до фонових елементів зображення у вигляді кривих ліній. Тобто, обрамлення тестового зображення має рисунок у формі кривих ліній, які перетинають в оригінальному зображенні слова “dog”, “collar” та “to”. За результатами виявлено, що розроблений алгоритм не сприймає ці додаткові елементи і не спотворює відтворений текст (рис.2.9). Можна прогнозувати, що розпізнавання символів проходить циклічні етапи перевірки з відкиданням зайвих елементів зображення. Натомість відмічено, що алгоритм не коректно працює з деякими символами, насамперед це стосується символів типу “лапки”. Крім цього, створений алгоритм має і свої обмеження. Так, він не коректно проводить розпізнавання літер не латинського алфавіту. Вочевидь, це обумовлено особливостями роботи модуля оптичного розпізнавання за допомогою бібліотеки pytesseract.

3. Визначено структуру електронної системи безпеки з проведенням вибору основного обладнання. Наведено особливості застосування моделі YOLO v7 для вирішення задачі з розпізнавання спеціальних об’єктів, до числа котрих віднесено холодну зброю. Розглянуто базову архітектуру нейронної мережі, яка визначає модель YOLO v.7 та її модифікована нова версія з додаванням капсульного шару. Проведена перевірка спроможності цих двох моделей на розпізнавання додаткового класу об’єктів. Для цього, сформовано спеціалізований навчальний набір з 2500 зображень та проведено навчання нейронної мережі. Для розпізнавання спеціального об’єкту, до етапу навчання додано новий 81 клас до стандартної класифікації моделі COCO, і збільшено глибину та ширину мережі (параметр *depth multiple* збільшено з 0,33 до 0,5; параметр *width multiple* збільшено з 0,5 до 0,75) та додано модуль (програмний механізм) уваги CBAM.

За результатами проведеного експерименту виявлено, що базова модель YOLO v7 не завжди ідентифікує об’єкти, які між особою перетинаються або при фіксації змінюють свою форму. Натомість, її запропонована модифікована

версія моделі YOLOv7 працює більш точно, адже вона виявила більше об'єктів на тестовому зображенні (5 об'єктів проти 6). Крім цього, отримано, що модифікована модель працює швидше. Так, для базової версії загальний час роботи складає 0,726 сек, тоді як при змінній архітектурі він виявився 0,575 сек.

4. Проведена практична апробація зміненої моделі на розпізнаванні об'єктів холодної зброї і отримано кращі результати з значень метрики mAP , у порівнянні з базовою архітектурою. Наприклад, при розпізнаванні об'єкту “ніж” модель YOLO v7 при відсутності капсульного шару вже не може провести розпізнавання об'єкту на визначеному пороговому рівні 50% на відстані 3 м, коли відсутнє освітлення за умов повного відображення об'єкту (табл.3.7). Наявність ж капсульного шару в структурі нейронної мережі моделі YOLO v7 за цих умов дозволила підвищити точність розпізнавання з 48% до 56%. Проведено додатково розрахунок метрики $F1$ і знайдено, що при звичайних умовах фіксації зображення камерою електронної системи безпеки оцінка ефективності модифікованої моделі YOLOv7 за рахунок додавання капсульного шару збільшується з 82% до 89%, що додає більшої довіри до отриманих результатів практичного експерименту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Varfolomieiev A., Antonyuk O., Lysenko O. CamShift Object Tracking Algorithm Implementation on DM6437 EVM // *Proceedings of the 4th European DSP in Education and Research Conference*. – 2010. – P. 96–100. – ISBN 978-0-9552047-5-3.
2. Horn B. K. P., Schunck B. G. Determining Optical Flow // *Artificial Intelligence*. – 1981. – Vol. 17, No. 1-3. – P. 185–203. – DOI: 10.1016/0004-3702(81)90024-2.
3. Lucas B., Kanade T. An Iterative Image Registration Technique with an Application to Stereo Vision // *Proceedings of Imaging Understanding Workshop*. – 1981. – Vol. 4. – P. 121–130.
4. Isik S. A Comparative Evaluation of Well-Known Feature Detectors and Descriptors // *International Journal of Applied Mathematics Electronics and Computers*. – 2015. – Vol. 3, No. 1. – P. 1–6. – DOI: 10.18100/ijamec.60004.
5. Bradski G. R. Computer Vision Face Tracking for Use in a Perceptual User Interface // *Intel Technology Journal*. – 1998. – Q2.
6. Jung E. J., et al. Development of a Laser-Rangefinder-Based Human Tracking and Control Algorithm for a Marathoner Service Robot // *IEEE/ASME Transactions on Mechatronics*. – 2014. – Vol. 19, No. 6. – P. 1963–1976. – DOI: 10.1109/TMECH.2013.2294180.
7. Redmon J., Farhadi A. YOLOv3: An Incremental Improvement // arXiv preprint arXiv:1804.02767. – 2018. – 28 p.
8. Gupta A., et al. Tracking Occluded Objects Using Deep Learning // *Journal of Visual Communication and Image Representation*. – 2021. – DOI: 10.1016/j.jvcir.2021.103225.
9. Zhang K., Zuo W., Chen Y., et al. Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising // *IEEE Transactions on Image Processing*. – 2017. – Vol. 26, No. 7. – P. 3142–3155.

- 10.Carletti V., Greco A., Saggese A., Vento M. Multi-Object Tracking by Flying Cameras Based on a Forward-Backward Interaction // *IEEE Access*. – 2018. – Vol. 6. – P. 43905–43919. – DOI: 10.1109/ACCESS.2018.2864672.
- 11.Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 2017.
- 12.Bradski G., Kaehler A. Learning OpenCV: Computer Vision with the OpenCV Library. – *Sebastopol: O'Reilly Media*, 2008. – 555 p.
- 13.Szeliski R. Computer Vision: Algorithms and Applications. – *Springer*, 2010. – 812 p. – DOI: 10.1007/978-1-84882-935-0.
- 14.Zhu M., et al. Robust Object Tracking Using Online Feature Selection // *IEEE Transactions on Multimedia*. – 2014. – Vol. 16, No. 3. – DOI: 10.1109/TMM.2013.2284891.
- 15.Zhang F. L., et al. Coherent Video Generation for Multiple Hand-Held Cameras with Dynamic Foreground // *Computational Visual Media*. – 2020. – Vol. 6, No. 3. – P. 291–306. – DOI: 10.1007/s41095-020-0187-3.
- 16.Mallat S. A Theory for Multiresolutional Signal Decomposition: The Wavelet Representation // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 1989. – Vol. 7. – P. 674–693.
- 17.Daubechies I. Ten Lectures on Wavelets. – *SIAM*, 1992. – DOI: 10.1137/1.9781611970104.
- 18.Lal S., Kumari P. Wavelet Approximation of Functions Belonging to Generalized Lipschitz Class by Extended Haar Wavelet Series // *International Journal of Applied and Computational Mathematics*. – 2019. – Vol. 5, No. 67. – DOI: 10.1007/s40819-019-0646-7.
- 19.Candes E. J., Donoho D. L. Curvelets: A Surprisingly Effective Nonadaptive Representation for Objects with Edges // *Curves and Surfaces*. – 2000. – DOI: 10.1007/978-1-4757-3124-8_5.
- 20.Otsu N. A Threshold Selection Method from Gray-Level Histograms // *IEEE Transactions on Systems, Man, and Cybernetics*. – 1979. – Vol. 9. – P. 62–66.

21.Shafait F., Keysers D., Breuel T. M. Efficient Implementation of Local Adaptive Thresholding Techniques Using Integral Images // *Document Recognition and Retrieval XV*. – 2008. – DOI: 10.1117/12.767755.

22.Smith R. An Overview of the Tesseract OCR Engine // *Proceedings of the Ninth International Conference on Document Analysis and Recognition*. – 2007. – Vol. 2. – P. 629–633.

23.Ul-Hasan A., et al. A Sequence Learning Approach for Scene Text Recognition // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 2019. – DOI: 10.1109/TPAMI.2019.2935780.

24.Murugesan K., Balasubramani P., Murugan P. R. A Quantitative Assessment of Speckle Noise Reduction in SAR Images Using TLFFBP Neural Network // *Arabian Journal of Geosciences*. – 2020. – Vol. 13, No. 35.

25.Gowthami S., Harikumar R. Conventional Neural Network for Blind Image Blur Correction Using Latent Semantics // *Soft Computing*. – 2020. – Vol. 24. – P. 15223–15237.

26.Kumar R., et al. Noise Reduction in Images Using Advanced Filtering Techniques // *IEEE Access*. – 2020. – Vol. 8. – P. 132134–132145. – DOI: 10.1109/ACCESS.2020.3015289.

27.Chen Y. Deep Learning for Image Denoising: A Review // *Computer Vision and Image Understanding*. – 2023. – DOI: 10.1016/j.cviu.2023.103687.

28.Qian Y., Dong J., Wang W., et al. Feature Learning for Steganalysis Using Convolutional Neural Networks // *Multimedia Tools and Applications*. – 2018. – Vol. 77. – P. 19633–19657.

29.Lee J., et al. Scene Parsing with Contextual Features for Object Recognition // *Journal of Pattern Recognition*. – 2019. – Vol. 95. – P. 12–25. – DOI: 10.1016/j.patcog.2019.04.023.

30.Li Z. Image Denoising Based on Deep Learning // *Highlights in Science Engineering and Technology*. – 2023. – Vol. 39. – P. 1245–1251.

31.Smith B. Advances in Object Recognition in Complex Environments // *Journal of Artificial Intelligence Research*. – 2022. – DOI:

10.1016/j.jair.2022.101112. A Computational Approach to Edge Detection // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 1986. – Vol. 8, No. 6. – P. 679–698.

32.Canny J. A Computational Approach to Edge Detection // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 1986. – Vol. 8, No. 6. – P. 679–698.

33.Sobel I., Feldman G. A 3x3 Isotropic Gradient Operator for Image Processing // *Stanford Artificial Intelligence Project*. – 1968.

34.Krizhevsky A., Sutskever I., Hinton G. E. Imagenet Classification with Deep Convolutional Neural Networks // *Advances in Neural Information Processing Systems*. – 2012.

35.Wei W., Cheng Y., He J. A Review of Small Object Detection Based on Deep Learning // *Neural Computing & Applications*. – 2024. – Vol. 36. – P. 6283–6303. – DOI: 10.1007/s00521-024-09422-6.

36.Tan L., Liu Z., Huang X. A Transformer-Based UAV Instance Segmentation Model TF-YOLOv7 // *SIViP*. – 2024. – Vol. 18. – P. 3299–3308.

37.Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Object Detection // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. – 2016.

38.Cai K., Yang J., Ren J. A Lightweight Algorithm for Small Traffic Sign Detection Based on Improved YOLOv5s // *SIViP*. – 2024. – Vol. 18. – P. 4821–4829. – DOI: 10.1007/s11760-024-03118-z.

39.Bochkovskiy A., Wang C., Liao H. YOLOv4: Optimal Speed and Accuracy of Object Detection // *arXiv preprint arXiv:2004.10934*. – 2020.

40.Jocher G. YOLOv5 by Ultralytics // *GitHub Repository*. – 2020. – URL: <https://github.com/ultralytics/yolov5> (дата звернення: 29.12.2025).

41.Mei T., Li J. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications // *arXiv preprint arXiv:2209.05687*. – 2022.

42.Jocher G., Chaurasia A., Stoken A., Borovec J. YOLO by Ultralytics // *arXiv preprint arXiv:2408.15857*. – 2024. – URL: <https://arxiv.org/abs/2408.15857> (дата звернення: 03.12.2024).

43.Kim H., Zhang W. YOLOv9: Revolutionizing Real-Time Object Detection // *arXiv preprint arXiv:2409.07813*. – 2024. – URL: <https://arxiv.org/abs/2409.07813> (дата звернення: 03.12.2024).

44.Liu W., Shen Z., Xu S. CF-YOLO: a Capable Forest Fire Identification Algorithm Founded on YOLOv7 Improvement // *SIViP*. – 2024. – DOI: 10.1007/s11760-024-03288-w.

45.Zhao D., Shao F., Yang L. Object Detection Based on an Improved YOLOv7 Model for UAV Patrol Tasks in Controlled Areas // *Electronics*. – 2023. – Vol. 12, No. 23. – DOI: 10.3390/electronics12234887.

46.Redmon J., Farhadi A. YOLO9000: Better, Faster, Stronger // *IEEE Conference on Computer Vision and Pattern Recognition*. – 2017.

47.Bewley A., Ge Z., Ott L., et al. Simple Online and Realtime Tracking with a Deep Association Metric // *IEEE International Conference on Image Processing (ICIP)*. – 2016.

48.He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition // *IEEE Conference on Computer Vision and Pattern Recognition*. – 2016.

49.Zhang S., Huang Z., Wang J. Automatic License Plate Recognition (ALPR) Based on Deep Learning Techniques // *IEEE Access*. – 2020. – Vol. 8.

50.Wu Y., Tan L., Zhang X. Real-Time Traffic Flow Prediction Using a Hybrid Deep Learning Model Based on CNN and LSTM // *Journal of Intelligent Transportation Systems*. – 2021.

51.Fu X., Wei G., Yuan X. Efficient YOLOv7-Drone: An Enhanced Object Detection Approach for Drone Aerial Imagery // *Drones*. – 2023.

52.Cho K., van Merriënboer B., Bahdanau D., Bengio Y. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches // *arXiv preprint arXiv:1409.1259*. – 2014.

53.Schroff F., Kalenichenko D., Philbin J. FaceNet: A Unified Embedding for Face Recognition and Clustering // *IEEE Conference on Computer Vision and Pattern Recognition*. – 2015.

54.Greff K., Srivastava R.K., Koutník J., Steunebrink B.R., Schmidhuber J. LSTM: A Search Space Odyssey // *IEEE Transactions on Neural Networks and Learning Systems*. – 2017.

55.Samek W., et al. Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models // *IEEE Signal Processing Magazine*. – 2017.

56.Damotharasamy S. Approach to model human appearance based on sparse representation for human tracking in surveillance // *IET Image Processing*. 2020. Vol. 14, No. 11. P. 2383–2394. DOI: 10.1049/iet-ipr.2018.5961.

57.Abughalieh K.M., Alawneh S.G. Predicting Pedestrian Intention to Cross the Road // *IEEE Access*. 2020. Vol. 8. P. 72558–72569. DOI: 10.1109/ACCESS.2020.2987777.

58.Li D., et al. A multi-type features method for leg detection in 2-D laser range data // *IEEE Sensors Journal*. 2017. Vol. 18, No. 4. P. 1675–1684. DOI: 10.1109/JSEN.2017.2784900.

59.Переверзев О.А., Трапезон К.О. Дослідження програмних алгоритмів відстеження руху об'єктів в електронних системах безпеки // *Вчені записки ТНУ імені В.І.Вернадського. Серія: Технічні науки*. – 2022. – Том 33(72), №6. – С. 288-293. – DOI: <https://doi.org/10.32782/2663-5941/2022.6/47>.

60.Nguyen V.T., Nguyen A.T., Nguyen V.T., Bui H.A., Nguyen X.T. Real-time Target Human Tracking using Camshift and Lucas-Kanade Optical Flow Algorithm // *Advances in Science, Technology and Engineering Systems Journal*. – 2021. – Vol. 6, No. 2. – P. 907–914. – DOI: 10.25046/aj0602103.

61.Hajipour K., Mehrdad V. Edge detection of noisy digital image using optimization of threshold and self-organized map neural network // *Multimed Tools Appl*. 2021. Vol. 80. P. 5067–5086. DOI: <https://doi.org/10.1007/s11042-020-09942-y>.

62.Kalare K., Bajpai M., Sarkar S., et al. Deep neural network for beam hardening artifacts removal in image reconstruction // *Appl. Intell.* 2022. Vol. 52. P. 6037–6056. DOI: <https://doi.org/10.1007/s10489-021-02604-y>.

63.Kaji S., Kida S. Overview of image-to-image translation by use of deep neural networks: denoising, super-resolution, modality conversion, and reconstruction in medical imaging // *Radiol. Phys. Technol.* 2019. Vol. 12. P. 235–248. DOI: <https://doi.org/10.1007/s12194-019-00520-y>.

64.Singh P., Shankar A. A novel optical image denoising technique using convolutional neural network and anisotropic diffusion for real-time surveillance applications // *J. Real-Time Image Proc.* 2021. Vol. 18. P. 1711–1728. DOI: <https://doi.org/10.1007/s11554-020-01060-0>.

65.Переверзев О., Трапезон К. Особливості використання нейронних мереж у задачах зменшення шуму цифрових зображень // *Вісник КрНУ імені Михайла Остроградського*. – 2023. - №3 (140). – С. 44-51. – DOI: <https://doi.org/10.32782/1995-0519.2023.3.6>.

66.Joshi K., Yadav R., Allwadhi S. PSNR and MSE based investigation of LSB // *International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*. 2016.

67.Bhathena J. Weather Dataset [Електронний ресурс] // *Kaggle*. – 2021. – Режим доступу: <https://www.kaggle.com/datasets/jehanbhathena/weather-dataset> (дата звернення: 18.12.2024).

68.Li D., Dai X., Gui J., et al. A reversible watermarking for image content authentication based on wavelet transform // *SIViP*. 2024. DOI: <https://doi.org/10.1007/s11760-023-02950-z>.

69.Chaibi N., Ben Aoun N., Eladel A., et al. Image super resolution boosting using beta wavelet // *SIViP*. 2023. DOI: <https://doi.org/10.1007/s11760-023-02887-3>.

70.Preetha S., Bindu V. A Wavelet Optimized Video Copy Detection Using Content Fingerprinting // *J. Sign. Process. Syst.* 2023. Vol. 95. P. 363–377. DOI: <https://doi.org/10.1007/s11265-022-01830-y>.

71. Chaudhuri A., Shivakumara P., Chowdhury P., et al. A deep action-oriented video image classification system for text detection and recognition // *SN Appl. Sci.* 2021. Vol. 3. P. 838. DOI: <https://doi.org/10.1007/s42452-021-04821-z>.

72. Manjunath Aradhya V., Basavaraju H., Guru D. Decade research on text detection in images/videos: a review // *Evol. Intel.* 2021. Vol. 14. P. 405–431. DOI: <https://doi.org/10.1007/s12065-019-00248-z>.

73. Daubechies I. The wavelet transform, time-frequency localization and signal analysis // *IEEE Trans. Inf. Theory*. 1990. Vol. 36, No. 5. P. 961–1005. DOI: <https://doi.org/10.1109/18.57199>.

74. Ramakrishnan V., Pete D. Haar Wavelet-Based Fusion of Multiple Exposure Images for High Dynamic Range Imaging // *SN Comput. Sci.* 2022. Vol. 3. P. 129. DOI: <https://doi.org/10.1007/s42979-021-01010-y>.

75. Переверзєв О., Трапезон К. Програмні підходи використання вейвлет-перетворення в задачах розпізнавання тексту із зображень // *Вісник КрНУ імені Михайла Остроградського*. – 2024. - №1 (144). – С. 266-272. – DOI: <https://doi.org/10.32782/1995-0519.2024.1.35>.

76. Li X., Shen Z. Driver fatigue detection based on improved YOLOv7 // *J. Real-Time Image Proc.* 2024. Vol. 21, No. 75. DOI: [10.1007/s11554-024-01455-3](https://doi.org/10.1007/s11554-024-01455-3).

77. Zhao D., Shao F., Yang L. Object detection based on an improved YOLOv7 model for unmanned aerial-vehicle patrol tasks in controlled areas // *Electronics*. 2023. Vol. 12, No. 23. P. 4887. DOI: [10.3390/electronics12234887](https://doi.org/10.3390/electronics12234887).

78. Zhou W., Cai C., Zheng L. ASSD-YOLO: a small object detection method based on improved YOLOv7 for airport surface surveillance // *Multimed. Tools Appl.* 2024. Vol. 83. P. 55527–55548. DOI: [10.1007/s11042-023-17628-4](https://doi.org/10.1007/s11042-023-17628-4).

79. Xu S., Chen Z., Zhang H. Improved remote sensing image target detection based on YOLOv7 // *Optoelectron. Lett.* 2024. Vol. 20. P. 234–242. DOI: [10.1007/s11801-024-3-z](https://doi.org/10.1007/s11801-024-3-z).

80. Li C., Wang J. Remote sensing image location based on improved YOLOv7 target detection // *Pattern Anal. Applic.* 2024. Vol. 27, No. 50. DOI: [10.1007/s10044-024-01276-x](https://doi.org/10.1007/s10044-024-01276-x).

81.Переверзєв О., Трапезон К. Принципи та підходи використання алгоритму YOLO v7 в системах виявлення об'єктів // *Вісник КрНУ імені Михайла Остроградського*. – 2024. - №4 (147). – С. 104-112. – DOI: <https://doi.org/10.32782/1995-0519.2024.4.13>.

82.Sun Y., Li Y., Li S. PBA-YOLOv7: An object detection method based on an improved YOLOv7 network // *Appl. Sci.* 2023. Vol. 13, No. 18. P. 10436. DOI: 10.3390/app131810.

83.Viola P., Jones M. Rapid object detection using a boosted cascade of simple features // *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. 2001. Vol. 1. P. 511–518. DOI: 10.1109/CVPR.2001.990517.

84.Hochreiter S., Schmidhuber J. Long short-term memory // *Neural Computation*. 1997. Vol. 9, No. 8. P. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

85.Wang C.-Y., Bochkovskiy A., Liao H.-Y. M. Scaled-YOLOv4: Scaling cross stage partial network // *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021. P. 13029–13038.

86.Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C.-Y., Berg A. C. SSD: Single shot multibox detector // *European Conference on Computer Vision*. 2016. P. 21–37.

87.Jocher G., Chaurasia A., Qiu J. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors // *arXiv preprint*. 2022. arXiv:2207.02696.

88.2.0MP IP-відеокамера BALTER IP-MT1210GR [Електронний ресурс] // *NeoLight*. – Режим доступу: https://neolight.in.ua/uk/videonabludenie/kamera_videonabludenia/2-0mp-ip-videokamera-balter-ip-mt1210gr (дата звернення: 02.12.2024).

89.SG72860M 24-портовий комутатор [Електронний ресурс] // *Versitron*. – Режим доступу: <https://www.versitron.com/products/sg72860m-24ports-10-100-1000-4-100-1g-sfp-managed> (дата звернення: 02.12.2024).

90.NUOO Програма IP [Електронний ресурс] // *Delishes*. – Режим доступу: <https://delishes.com.ua/nuuo-programma-ip.html> (дата звернення: 02.12.2024).

91.AJAX NVR-16-CH [Електронний ресурс] // *Ajax Systems*. – Режим доступу: <https://ajax.systems.ua/products/specs/nvr-16-ch/> (дата звернення: 02.12.2024).

92.Hikvision. [Електронний ресурс] // *Hikvision Surveillance Solutions*. 2023. URL: <https://www.hikvision.com> (дата звернення: 29.12.2024).

93.Dahua Technology. [Електронний ресурс] // *Dahua Security Systems Overview*. 2023. URL: <https://www.dahuasecurity.com> (дата звернення: 29.12.2024).

94.Balter Security. [Електронний ресурс] // *IP-MT1210GR Camera Specifications*. 2023. URL: <https://www.balter.com> (дата звернення: 29.12.2024).

95.Versitron. [Електронний ресурс] // *SG72860M Hybrid Optical Switch Specifications*. 2023. URL: <https://www.versitron.com> (дата звернення: 29.12.2024).

96.GitHub LabelImg [Електронний ресурс] / *Heartex Labs*. – Режим доступу: <https://github.com/heartexlabs/labelImg> (дата звернення: 02.12.2024).

97.Zhang Z., Liu Q., Wang Y. Deep learning-based object detection for remote sensing images: A survey // *IEEE Geosci. Remote Sens. Mag.* 2020. Vol. 8, No. 4. P. 16–38.

98.Li X., Chen W., Qiao Y. Common object detection in remote sensing images using a CNN-based model // *Remote Sens.* 2019. Vol. 11, No. 15. P. 1818.

99.Smith B., Zhang H. Improving Object Detection under Low-Light Conditions: A Review // *International Conference on Computer Vision*. – 2021.

100. Han X., Chen Y., Zhang J. Infrared small-target detection using deep learning: A survey // *IEEE Trans. Geosci. Remote Sens.* 2023.

101. Li Q., Zhang Y., Sun G. Enhancing infrared object detection via domain adaptation techniques // *Pattern Recognit. Lett.* 2024.

102. Sabour S., Frosst N., Hinton G. E. Dynamic routing between capsules // *Advances in Neural Information Processing Systems*. 2017.

103. Lalonde R., Khosravan N., Bagci U. Deformable Capsules for Object Detection // *arXiv preprint*. 2021. arXiv:2104.05031.
104. Hinton G. E., Sabour S., Frosst N. Matrix capsules with EM routing // *arXiv preprint*. 2018. arXiv:1710.09829.
105. Zhao W., Wang L., Yan Y. A survey on capsule networks for object recognition // *IEEE Trans. Neural Netw. Learn. Syst.* 2021.
106. Huang, J., Chang, Y., Li, W., Tong, J., & Du, S. (2024). A Spatio-Temporal Capsule Neural Network with Self-Correlation Routing for EEG Decoding of Semantic Concepts of Imagination and Perception Tasks. *Sensors*, 24(5988). DOI: 10.3390/s24185988.
107. Xiang C., Zhang L., Tang Y., Zou W., Xu C. MS-CapsNet: A Novel Multi-Scale Capsule Network // *IEEE Signal Processing Letters*. – 2018. – Vol. 25, No. 12. – P. 1850-1854. – DOI: 10.1109/LSP.2018.2873892.
108. Назаркевич М. А., Олексів Н. Т., Система розпізнавання об'єктів на основі моделі YOLO // *Український журнал інформаційних технологій*. 2024. Vol. 6, No. 1. P. 120–126. – DOI: <https://doi.org/10.23939/ujit2024.01.120>

ДОДАТОК А

Визначення тексту на складному фоні

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
import pywt
import pytesseract

# Встановлення шляху до Tesseract OCR
pytesseract.pytesseract.tesseract_cmd = r'C:\Program
Files\Tesseract-OCR\tesseract.exe'

# Завантаження зображення
image = cv2.imread('3.jpg', 0)      # Зчитування у
відтінках сірого

# Застосування порогової обрізки для виділення тексту
_, thresholded_image = cv2.threshold(image, 0, 255,
cv2.THRESH_BINARY + cv2.THRESH_OTSU)

# Вейвлет-перетворення зображення (використовуємо
вейвлет Хаара)
coeffs = pywt.dwt2(thresholded_image, 'haar')

# Реконструкція зображення з коефіцієнтів
reconstructed_image = pywt.idwt2(coeffs, 'haar')

# Перетворення зображення на цілочисельний тип
reconstructed_image = np.uint8(reconstructed_image)
```

```
# Розпізнавання тексту за допомогою pytesseract
recognized_text =
pytesseract.image_to_string(reconstructed_image)

# Відображення результатів та розпізнаного тексту
plt.figure(figsize=(12, 6))

plt.subplot(231)
plt.title('Original Image')
plt.imshow(image, cmap='gray')

plt.subplot(232)
plt.title('Thresholded Image')
plt.imshow(thresholded_image, cmap='gray')

plt.subplot(233)
plt.title('Reconstructed Image')
plt.imshow(reconstructed_image, cmap='gray')

plt.tight_layout()
plt.show()

# Виведення розпізнаного тексту

print("Recognized Text:")
print(recognized_text)
```

ДОДАТОК Б

1. Завантаження та підготовка датасету

```

```python
import os
import cv2
import numpy as np
from yolov7.utils import augment_data

Шлях до датасету
dataset_path = "/path/to/dataset/"

def load_images(dataset_path):
 images, labels = [], []
 for img_name in os.listdir(dataset_path):
 img_path = os.path.join(dataset_path,
img_name)

 label = 1 if "knife" in img_name else 0
 image = cv2.imread(img_path)
 image = cv2.resize(image, (640, 640))
 images.append(image)
 labels.append(label)
 return np.array(images), np.array(labels)

Завантаження даних
images, labels = load_images(dataset_path)

Аугментація даних
augmented_images, augmented_labels =
augment_data(images, labels)

```

```

 print(f"Датасет містить {len(augmented_images)}
зображень після аугментації.")
 ...

```

## 2. Навчання YOLOv7

```

```python
from yolov7 import YOLOv7

# Ініціалізація моделі
model = YOLOv7(config_path='yolov7.cfg',
weights_path='yolov7.weights', device='cuda')

# Налаштування параметрів навчання
model.set_hyperparams(
    batch_size=16,
    epochs=50,
    learning_rate=0.001
)

# Початок навчання
model.train(train_data=augmented_images,
val_data=None)

# Збереження навченої моделі
model.save("yolov7_trained.pth")
...

```

3. Тестування моделі

```

```python
from yolov7.utils import evaluate_model

```

```

Оцінка моделі на різних умовах
conditions = [
 {"distance": 1, "transparency": 1.0},
 {"distance": 2, "transparency": 1.0},
 {"distance": 5, "transparency": 0.5},
]

results = []
for cond in conditions:
 test_images = simulate_conditions(images,
distance=cond["distance"],
transparency=cond["transparency"])
 result = evaluate_model(model, test_images)
 results.append(result)
 print(f"Умови: {cond}, Точність:
{result['accuracy']:.2f}%")
 ...

```

#### 4. Аналіз відеопотоку в реальному часі

```

```python
import cv2

# Відкриття відеопотоку
cap = cv2.VideoCapture(0)

while True:
    ret, frame = cap.read()
    if not ret:
        break

# Детекція об'єктів

```

```

detections = model.detect(frame)

# Візуалізація
for det in detections:
    x1, y1, x2, y2, conf, cls = det
    cv2.rectangle(frame, (x1, y1), (x2, y2), (0,
255, 0), 2)
    cv2.putText(frame, f"{cls} ({conf:.2f})",
(x1, y1 - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255,
0), 2)



cv2.imshow("Відеопотік", frame)

if cv2.waitKey(1) & 0xFF == ord('q'):
    break

cap.release()
cv2.destroyAllWindows()
` ``

```

5. Зображення з результатами розпізнавання ножа при наявності освітлення моделлю YoloV7 при різних відстанях та умовах перекриття.

Умови, відстань (м), точність розпізнавання (%) без капсульного шару	Зображення
Повне відображення, 1 м, 87%	
Часткове перекриття, 1 м, 79%	

Повне відображення,
2 м, 83%



Часткове перекриття,
2 м, 75%



Повне відображення,
3 м, 71%



Часткове перекриття,
3 м, 69%



Повне відображення,
4 м, 59%



Часткове перекриття,
4 м, 48%






Повне відображення,
5 м, 33%



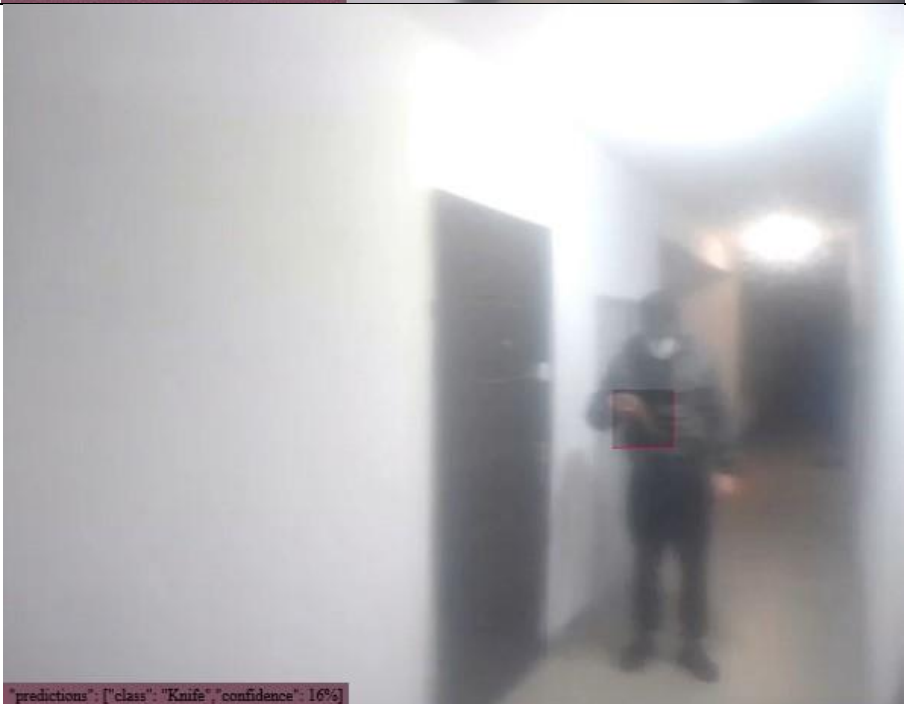




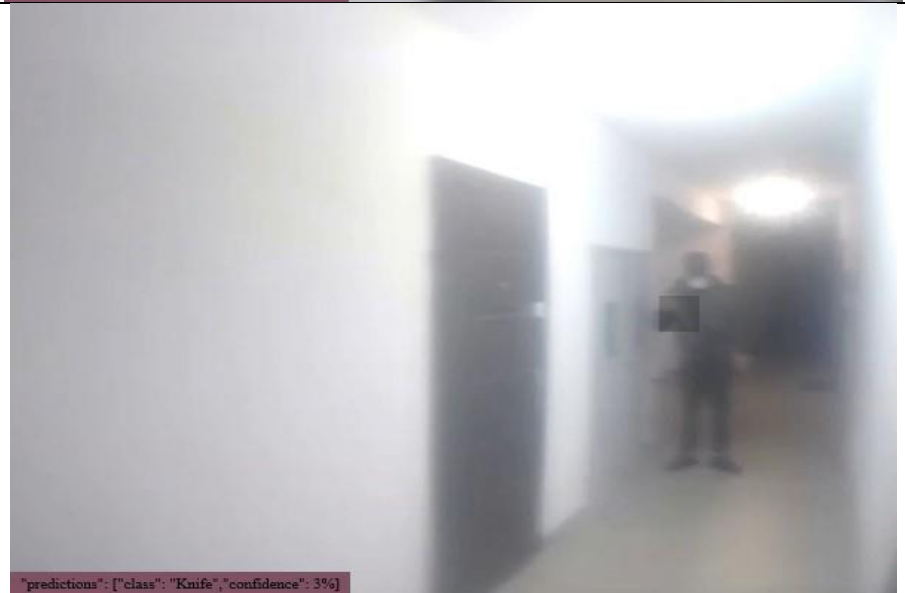
Часткове перекриття,
5 м, 18%




6. Зображення з результатами розпізнавання ножа при наявності освітлення та імітацією несприятливих умов моделлю YoloV7 при різних відстанях та умовах перекриття.


Умови, відстань (м), точність розпізнавання (%) без капсульного шару	Зображення
Повне відображення, 1 м, 53%	
Часткове перекриття, 1 м, 37%	
Повне відображення, 2 м, 32%	




<p>Часткове перекриття, 2 м, 28%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 28%}]</p>
<p>Повне відображення, 3 м, 20%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 20%}]</p>
<p>Часткове перекриття, 3 м, 16%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 16%}]</p>




<p>Повне відображення, 4 м, 9%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 9%}]</p>
<p>Часткове перекриття, 4 м, 6%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 6%}]</p>
<p>Повне відображення, 5 м, 3%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 3%}]</p>




Часткове перекриття, 5 м, 2%	 <p>"predictions": [{"class": "Knife", "confidence": 2%}]</p>
---------------------------------	---

7. Зображення з результатами розпізнавання ножа при відсутності освітлюваності моделлю YoloV7 при різних відстанях та умовах перекриття.

Умови, відстань (м), точність розпізнавання (%) без капсульного шару	Зображення
Повне відображення, 1 м, 81%	 <p>"predictions": [{"class": "Knife", "confidence": 81%}]</p>



<p>Часткове перекриття, 1 м, 67%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 67%}]</p>
<p>Повне відображення, 2 м, 77%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 77%}]</p>
<p>Часткове перекриття, 2 м, 61%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 61%}]</p>


<p>Повне відображення, 3 м, 50%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 50%}]</p>
<p>Часткове перекриття, 3 м, 22%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 22%}]</p>
<p>Повне відображення, 4 м, 15%</p>	 <p>"predictions": [{"class": "Knife", "confidence": 15%}]</p>

<p>Часткове перекриття, 4 м, 11%</p>	 <p>"predictions": ["class": "Knife", "confidence": 11%]</p>
<p>Повне відображення, 5 м, 9%</p>	 <p>"predictions": ["class": "Knife", "confidence": 9%]</p>
<p>Часткове перекриття, 5 м, 6%</p>	 <p>"predictions": ["class": "Knife", "confidence": 6%]</p>

8. Зображення з результатами розпізнавання ножа при відсутності освітлюваності та імітацією несприятливих умов моделлю YoloV7 при різних відстанях та умовах перекриття.

Умови, відстань (м), точність розпізнавання (%) без капсульного шару	Зображення
Повне відображення, 1 м, 32%	
Часткове перекриття, 1 м, 20%	

<p>Повне відображення, 2 м, 25%</p>	 <p>predictions: [{"class": "Knife", "confidence": 25%}]</p>
<p>Часткове перекриття, 2 м, 12%</p>	 <p>predictions: [{"class": "Knife", "confidence": 12%}]</p>
<p>Повне відображення, 3 м, 8%</p>	 <p>predictions: [{"class": "Knife", "confidence": 8%}]</p>

<p>Часткове перекриття, 3 м, 6%</p>	
<p>Повне відображення, 4 м, 4%</p>	