

<https://doi.org/10.31891/2307-5732-2025-357-71>

УДК 004.89

ОЛІЙНИК ІВАН

Хмельницький національний університет

<https://orcid.org/0009-0008-1890-9083>

e-mail: alyymazoku@gmail.com

МАНЗЮК ЕДУАРД

Хмельницький національний університет

<https://orcid.org/0000-0002-7310-2126>

e-mail: eduard.em.km@gmail.com

ПАСІЧНИК ОЛЕКСАНДР

Хмельницький національний університет

<https://orcid.org/0000-0002-8760-4688>

e-mail: o.a.pasichnyk@gmail.com

СКРИПНИК ТЕТЯНА

Хмельницький національний університет

<https://orcid.org/0000-0002-8531-5348>

e-mail: tkskripnik1970@gmail.com

МЕТОД РОЗПІЗНАВАННЯ ОБ'ЄКТІВ У РЕАЛЬНОМУ ЧАСІ ДЛЯ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ У ЗАВДАННЯХ РОЗСЛІДУВАНЬ І ПОШУКОВИХ

У роботі запропоновано метод розпізнавання об'єктів у реальному часі для безпілотних літальних апаратів у завданнях розслідувань і пошукових робіт. Відеопотік з БПЛА передається на наземну станцію, де за допомогою нейронних мереж типу YOLO здійснюється високоточне розпізнавання без необхідності зміни апаратного забезпечення. Розроблена інформаційна система дозволяє автоматизувати процес виявлення об'єктів та підвищити ефективність виконання оперативних завдань.

Ключові слова: безпілотний літальний апарат, комп'ютерний зір, розпізнавання об'єктів, нейронні мережі, реальний час, YOLO.

OLIINYK IVAN, MANZIUK EDUARD, PASICHNYK OLEKSANDR, SKRYPNYK TETIANA

Khmelnytskyi National University

A REAL-TIME OBJECT RECOGNITION METHOD FOR UNMANNED AERIAL VEHICLES IN INVESTIGATIVE AND SEARCH OPERATIONS

The rapid integration of unmanned aerial vehicles (UAVs) into civil, defense, and emergency response sectors has fundamentally transformed data collection and analysis methodologies in time-critical scenarios. UAVs have become indispensable tools in missions where human access is restricted, dangerous, or logistically challenging. Their capacity to capture high-resolution real-time aerial imagery makes them vital assets in search-and-rescue operations, post-disaster damage assessments, criminal investigations, and surveillance monitoring.

This research addresses the fundamental challenge of implementing efficient real-time object detection and recognition systems that operate with minimal human intervention while accommodating the inherent computational limitations of onboard UAV hardware. This study proposes an innovative approach leveraging external computational resources by transmitting UAV video feeds to ground-based processing systems where advanced YOLOv8 neural network models perform sophisticated real-time object detection and classification. The YOLOv8 architecture represents an optimal balance between processing speed and detection accuracy. This methodology significantly enhances recognition capabilities without requiring modifications to existing UAV hardware configurations, ensuring broad compatibility and cost-effectiveness. The system demonstrates remarkable performance metrics: preprocessing time averaging 22.5 ms, inference time of 5.5 ms, and visualization rendering within 3 ms per frame.

Multiple specialized datasets were utilized, including Top-View Drone Car Detection Dataset and YOLO HighVis Person Detection Dataset, ensuring robust multi-class object recognition capabilities. Experimental results conclusively confirm the technical and practical feasibility of seamlessly integrating AI-based detection systems into existing UAV monitoring infrastructures, achieving stable operation with accuracy fluctuations not exceeding 1% at 5m, 10% at 50m, and 35% at 100+ meters, thereby significantly enhancing the capabilities and effectiveness of future autonomous aerial surveillance platforms.

Keywords: UAV, real-time object detection, YOLO, computer vision, aerial surveillance, AI-based recognition, search and rescue.

Стаття надійшла до редакції / Received 01.07.2025

Прийнята до друку / Accepted 15.08.2025

Постановка проблеми

Застосування безпілотних літальних апаратів у сфері моніторингу, пошукових операцій і розслідувань стрімко зростає завдяки їхній мобільності та здатності отримувати дані з важкодоступних ділянок. Проте ефективність використання БПЛА безпосередньо залежить від здатності системи обробляти відеоінформацію в реальному часі та забезпечувати точне розпізнавання об'єктів [1], [2].

На сучасному етапі розвитку технологій штучного інтелекту та комп'ютерного зору, глибокі згорткові нейронні мережі демонструють високі результати у задачах детекції та класифікації об'єктів [3], проте їх застосування на борту БПЛА обмежується через дефіцит обчислювальних ресурсів. Як зазначено у [4], обробка відеопотоку з БПЛА за допомогою складних моделей на самому пристрої часто

призводить до затримок і зниження точності через необхідність спрощення моделей або зниження роздільної здатності кадрів.

Подальше вдосконалення систем розпізнавання об'єктів для БПЛА можливе при застосуванні підходу з передачею відео на наземну станцію для обробки потужнішими обчислювальними системами. У роботі [5] описано архітектуру з передачею відео в реальному часі та застосуванням моделі YOLOv5, яка забезпечує прийнятний компроміс між швидкістю обробки та точністю детекції. Дослідження [6] підтверджують доцільність використання моделей YOLO-сімейства у завданнях із жорсткими часовими обмеженнями, зокрема в умовах пошуково-рятувальних операцій.

Незважаючи на наявні напрацювання, залишається актуальним завдання створення методики розпізнавання об'єктів у реальному часі, що:

- працює зі зниженою затримкою;
- здатна до мультиоб'єктної детекції;
- стійка до змін умов освітлення, фону, ракурсу;
- не потребує модифікації апаратної частини БПЛА.

Таким чином, з огляду на вище сказане постає необхідність у розробці методу та програмної системи, що забезпечить ефективне, швидке та точне розпізнавання об'єктів у відеопотоці з БПЛА на основі сучасних технологій комп'ютерного зору та глибинного навчання.

Аналіз останніх джерел

Реалізація методу розпізнавання об'єктів у реальному часі для безпілотних літальних апаратів у завданнях розслідувань і пошукових робіт потребувала широкого спектра бібліотек для глибокого навчання, обробки відео та створення користувацького інтерфейсу. Однак поряд із технічними компонентами, не менш важливим є вивчення вже наявних рішень у цій сфері. Аналіз таких систем дозволяє зрозуміти, які підходи є ефективними, а також виявити слабкі місця, що вимагають удосконалення.

В роботі [7] наведено концепцію побудови мультимодальної системи виявлення об'єктів у складних погодних умовах. Це дозволяє підвищити точність детекції при слабкому освітленні або димі. У роботі [8] запропоновано використання моделі YOLOv7 для виявлення та звітності пошкодження доріг. Рішення дозволяло здійснювати виявлення потенційно небезпечних ділянок з висоти польоту БПЛА що було реалізовано під час публічних заходів у густонаселених районах. Такий підхід демонструє не лише технічну ефективність, а й соціальну значущість, адже системи виявлення критичних частин на ділянках автострадах із БПЛА здатні оперативно попереджати надзвичайні ситуації, сприяючи безпеці громадян. Інше джерело [9] пропонує метод використання PyQt5 для реалізації прозорого інтерфейсу над відеопотоком, що буде актуальним при розробці методу.

Розробка ефективного методу для виявлення об'єктів з БПЛА передбачає використання програмних засобів, що забезпечують як швидкодію, так і можливість масштабування. Ядром системи являється бібліотека PyTorch [10], яка реалізує високопродуктивну інфраструктуру для побудови та навчання нейронних мереж. Важливим компонентом є Ultralytics YOLOv8 [11], що надає готові інструменти для навчання, перевірки та експорту моделей у форматах різних форматів. Його інтеграція дозволяє реалізовувати детекцію в реальному часі навіть у системах із обмеженими ресурсами. У поєднанні з OpenCV [12] модель здатна швидко обробляти відео з БПЛА, виконувати масштабування, трансформацію, а також інтеграцію з зовнішніми потоками відео, що робить її придатною для широкого спектру задач, як охорони периметра та моніторингу надзвичайних ситуацій. Бібліотека NumPy [13] підтримує математичні обчислення в реальному часі, необхідні для роботи з великими масивами даних, особливо при візуалізації результатів. PyQt5 [14] використовується для створення прозорого інтерфейсу, що накладається на відеопотік, дозволяючи оператору оперативно взаємодіяти з системою. Додаткові бібліотеки, як-от pygetwindow [15] і pywin32 [16], дозволяють реалізувати повноцінну взаємодію з ОС, включаючи керування вікнами та перемикання між задачами.

Це відкриває можливості для автоматизації або розширення функціоналу у вигляді напівавтономних режимів роботи. Комплексне використання описаних інструментів формує базу для побудови адаптивного, масштабованого та гнучкого рішення, яке в подальшому може бути вдосконалене за рахунок нових архітектур, інтеграції зі сторонніми сенсорами або впровадження систем автономного прийняття рішень на основі штучного інтелекту. Це створює перспективу для застосування системи у складних умовах, зокрема в бойових, пошукових чи індустріальних середовищах, де точність та час реакції критично важливі.

Мета роботи: розробка ефективного методу розпізнавання об'єктів у реальному часі для безпілотних літальних апаратів, призначеного для завдань розслідувань і пошукових операцій, який забезпечує високу точність детекції при мінімальних затримках обробки без необхідності модифікації апаратного забезпечення БПЛА.

Виклад основного матеріалу

Метод розпізнавання об'єктів у реальному часі для безпілотних літальних апаратів у завданнях розслідувань і пошукових робіт потребує надходження відеоданих на ПК із бортової системи БПЛА, тим самим дані проходять подальшу обробку методом, що представлено нижче (рис. 1). У цій схемі детально відображено етапи методу виявлення об'єктів у відеопотоці, починаючи від захоплення

зображення до виведення результатів на екран. Такий підхід дозволяє системі ефективно аналізувати вхідні відеодані, забезпечуючи оперативну реакцію та високий рівень точності при виконанні завдань спостереження, пошуку або моніторингу обстановки.

Схема методу виявлення об'єктів передбачає надходження відеопотоку до системи з попередньо навченою моделлю YOLOv8. Обробка кадрів здійснюється з урахуванням заданих параметрів, а бібліотека mss забезпечує швидке захоплення зображення. Далі кадр аналізується моделлю, яка ідентифікує об'єкти, визначаючи їхні координати, клас і ймовірність. Результати візуалізуються за допомогою PyQt5, із відображенням підписів та рамок поверх відео. Система працює безперервно, оновлюючи кадри в миттєво.

Ще одним важливим фактором при розробці методу, є набір функціональних модулів, які призначені для обробки відеопотоку в реальному часі з використанням моделі YOLOv8, бібліотеки MSS для захоплення кадрів та технології оверлейної візуалізації на основі PyQt5. Запропонований метод реалізує локальну обробку відео на ПК. Архітектура системи охоплює шість основних модулів, які спільно забезпечують повний цикл роботи (рис. 2).

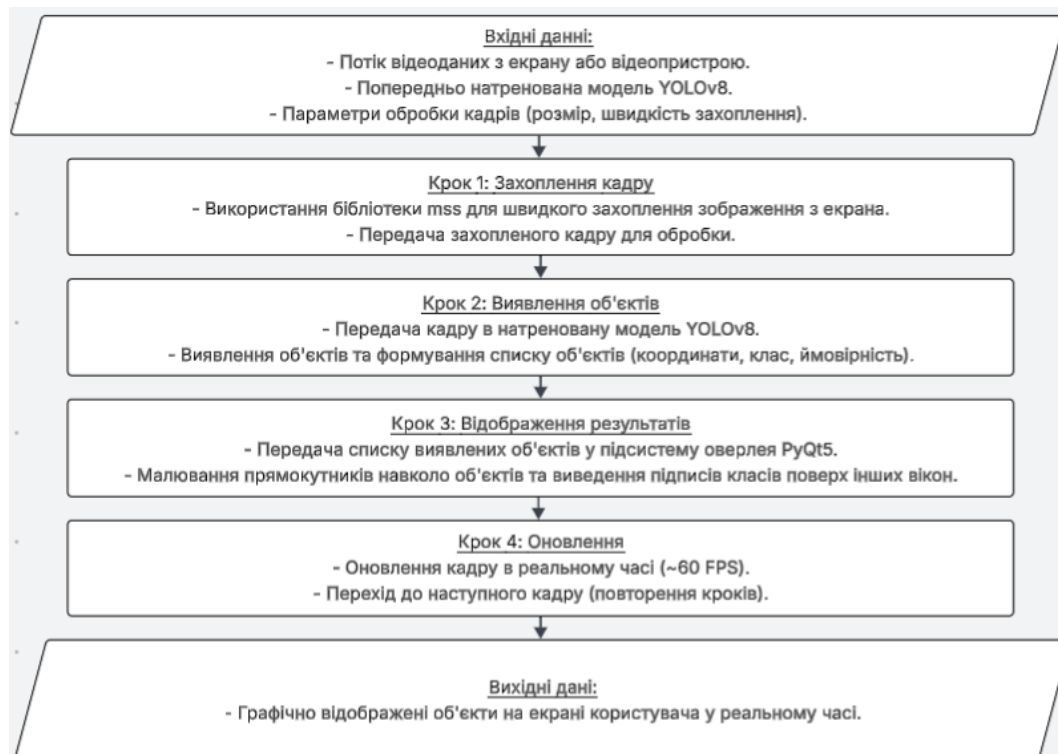


Рис. 1 Схема методу виявлення об'єктів на відеопотоці

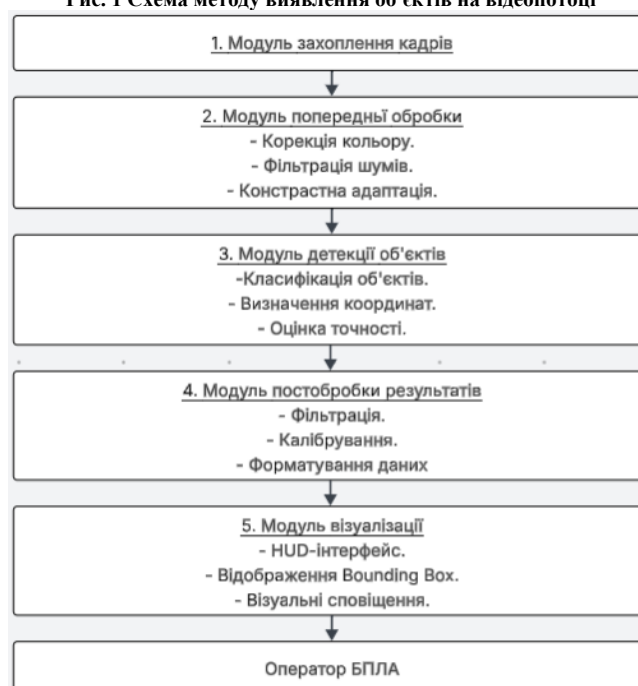


Рис. 2 Функціональні модулі обробки відеоданих БПЛА

Архітектура системи охоплює шість основних модулів, які спільно забезпечують повний цикл роботи (рис. 2):

- захоплення зображення: використовується MSS для зйомки з мінімальною затримкою та буферизацією;
- попередня обробка: корекція кольору, фільтрація шумів, підвищення контрасту;
- детекція об'єктів: розпізнавання за моделлю YOLOv8-nano з побудовою координат і визначенням класів;
- постобробка: фільтрація результатів, калібрування координат, підготовка даних;
- візуалізація: накладання графіки, підписів та елементів інтерфейсу на відео;
- інтерфейс оператора: сповіщення, метадані та елементи керування для зручності взаємодії.

Послідовність передачі відеоданих від БПЛА до методу озпізнавання об'єктів у реальному часі для безпілотних літальних апаратів у завданнях розслідувань і пошукових робіт є ключовою для ефективної обробки та виявлення об'єктів, що критично важливо для розслідувань і пошукових операцій. Запропоновані функціональні модулі методу, розроблені з урахуванням модульності, асинхронності та масштабованості, забезпечують низьку затримку, стабільну роботу при високому FPS та легку інтеграцію з різними типами БПЛА, зберігаючи при цьому основну логіку системи. Така архітектура гарантує високу ефективність і адаптивність до динамічних умов розпізнавання.

Звичайно, доцільно підкреслити, що в основі розробленого методу лежить архітектура YOLO, як вже говорилося раніше, вона є одною з найефективніших та найпоширеніших у сфері комп'ютерного зору. Саме завдяки цій архітектурі стало можливим досягти високої швидкості обробки кадрів у поєднанні з достатньою точністю, що є критично важливим для застосування в системах безпілотних літальних апаратів під час виконання завдань пошуку, моніторингу чи розслідувань. YOLOv8 вирізняється оптимізованою структурою, яка забезпечує ефективне вилучення ознак, глибoku багаторівневу обробку даних та швидку детекцію об'єктів з мінімальними затримками.

Як зображено нижче, архітектура YOLOv8 умовно поділяється на три основні частини: Backbone, Neck і Head. Кожен з цих блоків виконує окрему функцію в загальному процесі обробки зображення (рисунок 3). Модуль Backbone відповідає за вилучення первинних ознак з вхідного зображення. Він складається з кількох послідовних згорткових шарів та блоків C2f, що реалізують з'єднання ознак із використанням механізмів shortcut для збереження просторової інформації та забезпечення глибокої репрезентації ознак на різних рівнях. У межах цього блоку також відбувається поступове зменшення просторової роздільної здатності, що дозволяє фокусувати обчислювальні ресурси на важливих об'єктах.

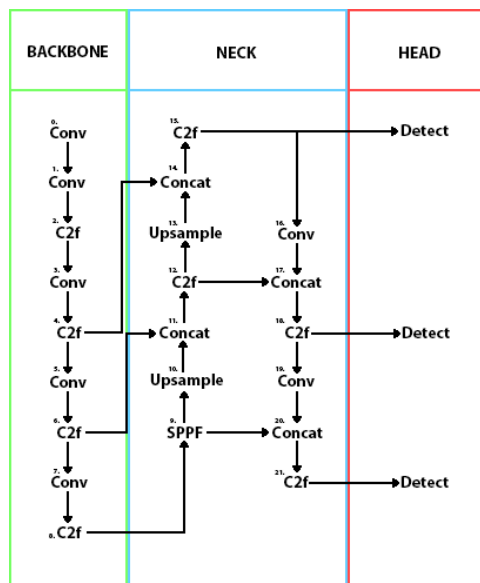


Рис. 3. Загальна архітектура YOLOv8 та її основні частини

Модуль Neck виконує проміжну обробку ознак і є зв'язувальним етапом між вилученням та детекцією. Головною цівіллю модуля Neck є об'єднання ознак з різних рівнів абстракції через операції Upsample та Concat, що дозволяє зберігати як детальні, так і узагальнені характеристики об'єктів. Також у Neck застосовується блок SPPF, що сприяє агрегації просторової інформації без збільшення обчислювальної складності.

Модуль Head відповідає безпосередньо за процес виявлення об'єктів на зображенні. На цьому етапі результати попередніх обчислень використовуються для побудови передбачень: визначення координат прямокутників, класів об'єктів та відповідних ймовірностей. Саме з цієї частини надходить вихідний список об'єктів для подальшої візуалізації або аналітичної обробки.

Архітектура YOLOv8 є справжнім проривом у сфері комп'ютерного зору, пропонуючи не просто ефективну, а й надзвичайно гнучку схему обробки зображень. Ця інноваційна розробка дозволяє безшовно інтегрувати її в найскладніші та найвідповідальніші завдання комп'ютерного зору, що виконуються БПЛА. Завдяки цьому досягається безпрецедентний рівень автоматизації виявлення об'єктів, що є критично важливим для багатьох застосувань, от як у нашому випадку моніторингу територій та пошуково-рятувальних операцій, особливо в умовах, коли кожна секунда на рахунок. При цьому, YOLOv8 здатна враховувати суттєві обмеження апаратного забезпечення, притаманні платформам БПЛА, де кожен ват потужності та кожен мегабайт пам'яті на рахунок. Вона оптимізована для обробки відеопотоку в режимі реального часу, забезпечуючи мінімальні затримки, що життєво необхідно для швидкого реагування та прийняття рішень у динамічному середовищі. Таким чином, YOLOv8 не просто виявляє об'єкти, а робить це з високою швидкістю та точністю, перетворюючи БПЛА на потужний інструмент для автономного моніторингу та аналізу ситуації, що є особливо актуальним сьогодні. Проте, слід пам'ятати, що без якісно зібраного та розміченого датасету вся ця висока ефективність згортається нанівець.

Ефективна робота розробленого методу розпізнавання об'єктів у реальному часі для безпілотних літальних апаратів у завданнях розслідувань і пошукових робіт на базі YOLOv8 вимагає стандартизованої структури проєкту. Усі дані організовано в єдину директорію data з підпапками train та valid, кожна з яких містить images та labels. Мітки об'єктів створюються у YOLO TXT форматі з нормалізованими координатами. Ця структура дозволяє автоматизувати зачитування даних, генерацію батчів та логічну перевірку, мінімізуючи втручання користувача. Вона також спрощує інтеграцію кількох датасетів для мультикласового або тонкого донавчання, забезпечуючи масштабованість. Для управління класами та шляхами використовується YAML-конфігураційний файл, що знижує кількість помилок та гарантує безперервність потоку даних від навчання до реального застосування.

Вибір та формування репрезентативного набору даних є ключовим етапом. Через значну варіативність об'єктів, ракурсів, освітлення та фону, було прийнято рішення використовувати кілька незалежних моделей YOLOv8, кожна з яких навчена на вузькоспеціалізованому датасеті. Це дозволяє досягти вищої точності та зменшити кількість хибнопозитивних спрацьовувань.

У рамках розробки сктоду використано щонайменше вісім окремих датасетів. Деякі з них були об'єднані для посилення варіативності в межах одного класу, інші використовувались автономно:

- Top-View Drone Car Detection Dataset [17];
- YOLO HighVis and Person Detection Dataset [18];
- Human Detector [19];
- Human DetectorJAAD Frames Dataset [20];
- Human detectorjaad frames datasetsMilitary Assets Dataset [21].

Для кожної моделі створювались окремі директорії train/ та valid/ з images/ та labels/, розмітка у форматі TXT. Анотація зображень частково виконувалась вручну, а валідація формувалась стратифіковано. Тренування окремих моделей дозволяє активувати лише потрібну модель залежно від конкретного завдання.

Реалізація методу розпізнавання об'єктів у реальному часі для безпілотних літальних апаратів у завданнях розслідувань і пошукових робіт вимагала використання широкого спектра програмних бібліотек, що забезпечують її функціональність на всіх етапах. Особливу увагу приділено бібліотекам для глибокого навчання, обробки зображень та відео, а також створення графічного інтерфейсу користувача. Для розробки методу застосовано такі ключові бібліотеки: PyTorch для запуску моделі та обробки зображень, Ultralytics як офіційна оболонка YOLOv8 для зручного інтерфейсу до нейронної мережі, OpenCV для захоплення та попередньої обробки відеокадрів і миттєвого відображення результатів.

Експериментальне дослідження методу

Програмна частина системи використовує стек технологій, що включає Ultralytics як інтерфейс до моделі YOLOv8, PyQt5 для користувацького інтерфейсу та OpenCV для роботи з відеопотоком. Завдяки модульній архітектурі, компоненти, такі як захоплення кадру, детекція об'єктів та візуалізація, можуть тестуватися незалежно.

Дослідження системи зосереджені на таких цілях:

- функціональне тестування: перевірка точності виявлення об'єктів у реальному часі;
- оцінка швидкодії;
- тестування стабільності: аналіз роботи під навантаженням без зниження якості чи збоїв;
- тестування адаптивності: визначення ефективності в різних умовах;
- тестування точності: оцінка ефективності визначення класу на різних відстанях.

Дослідження проводитимуться на симульованих відеопотоках з БПЛА, що максимально наближені до реальних польових умов. Ці відео охоплюють різноманітні сценарії, включаючи денне/нічне освітлення, мінливу погоду, динамічний рух та складні сцени, що дозволить перевірити стійкість, точність та адаптивність системи. Особлива увага буде приділена стабільності детекції при нестабільному русі камери та наявності візуальних перешкод, що є типовим для роботи БПЛА в умовах миттєвої обробки.

Усі дослідження виконувались з фіксованими параметрами, а результати ретельно документувалися. Першим тестуванням було функціональне, спрямоване на перевірку здатності системи точно виявляти об'єкти у відеопотоці в реальному часі. Окрім контрольних зразків, для тестування використовувалися відеоматеріали з реальних бойових зон, зокрема, із переміщенням військової техніки в умовах зруйнованої інфраструктури, що створювало складні умови. Особливу увагу приділяли розпізнаванню бойової одиниці M2 Bradley. Незважаючи на розмитість кадрів та візуальні перешкоди, система успішно виявила та надійно відстежувала транспортний засіб, підтверджуючи свою ефективність у критичних ситуаціях. Цей результат створює передумови для використання методу в системах тактичного аналізу та корегування дій у реальному часі (рис. 4). Також для контрольної перевірки було використано відеоряд із рухом важкої техніки під час бойових дій (рис. 5).



Рис. 4 Виявлення M2 Bradley



Рис. 5 – Виявлення важкої техніки

Окрім функціонального тестування, було проведено тестування швидкодії та стабільності розробленого методу в умовах, максимально наближених до реальної експлуатації. Аналіз показав, що етап попередньої обробки зображення займає в середньому 22.5 мс, безпосереднє розпізнавання об'єктів нейромережею близько 5.5 мс, а відображення результатів та побудова обмежувальних рамок займає 3 мс. Сукупний час обробки кадру залишається в межах, що забезпечують миттєву обробку без відчутної затримки (рис. 6).

Наступна діаграма ілюструє стабільність системи за параметром FPS у динаміці, залежно від кількості об'єктів у кадрі. Аналіз показав, що за відсутності об'єктів швидкодія системи досягає 40-41 кадр/с. При збільшенні кількості об'єктів FPS поступово знижується: для одного об'єкта він становить 28-31 кадр/с, для двох-трьох вже 25-28 кадр/с, а для трьох або більше об'єктів становила 18-20 кадрів/с. Попри зростання навантаження, система демонструє стабільну адаптивність, підтримуючи прийнятний рівень обробки без критичних втрат швидкодії, що дозволяє зберігати високу якість детекції (рис. 7).

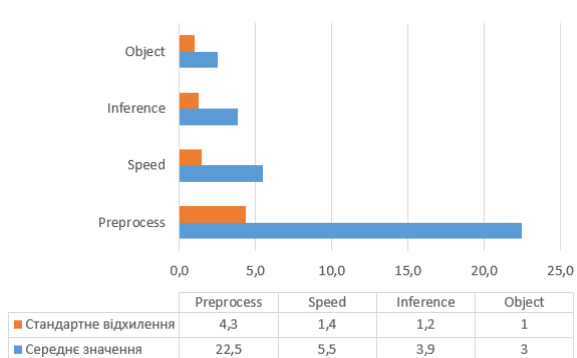


Рис. 6 Динаміка швидкодії методу

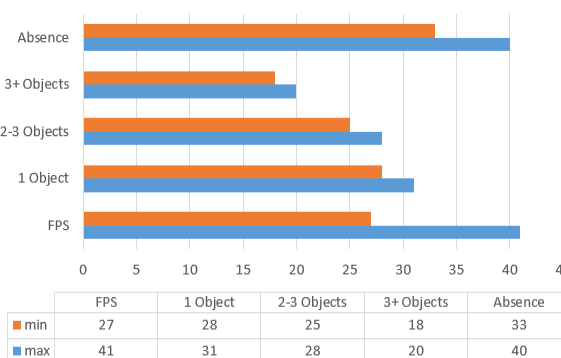


Рис. 7 Характеристика FPS

Одним із найскладніших тестів для систем комп'ютерного зору є перевірка їхньої адаптивності до умов зйомки, адже здатність розпізнавати об'єкти за різних обставин суттєво впливає на практичне застосування технології. Зважаючи на те, що об'єкти можуть перебувати як у відкритому полі, так і в урбанізованому середовищі під впливом численних перешкод, критично важливо забезпечити стабільність роботи системи незалежно від зовнішніх чинників. Саме тому додатково поведеться оцінки ефективності розробленої системи в умовах, наближених до реальних сценаріїв, було проведено тестування з використанням відеофрагментів із поліційних розслідувань. Такі матеріали, як правило, характеризуються обмеженою якістю, низьким контрастом, високим рівнем шуму та нестабільною зйомкою, що створює оптимальні умови для перевірки стійкості алгоритмів. Один із протестованих

відеоредактор ілюстрував ситуацію, в якій необхідно було здійснити пошук підозрюваної особи на основі наявного відеозапису. Незважаючи на розмитість зображення та слабе освітлення, модель успішно ідентифікувала людину з достатнім рівнем впевненості. Отримані результати підтвердили можливість застосування системи у сфері правоохоронної діяльності, де часто використовуються відеодокази з камер спостереження або портативних пристроїв із нестабільним потоком даних (рис. 8).



Рис. 8 Виявлення об'єкту person

Останнє тестування сфокусоване на точності розпізнавання об'єктів. Основна увага приділялася коректності визначення класу об'єкта на відстанях від 5 до понад 100 метрів, а також частоті зміни класу при використанні мультикласових датасетів (рис. 9).

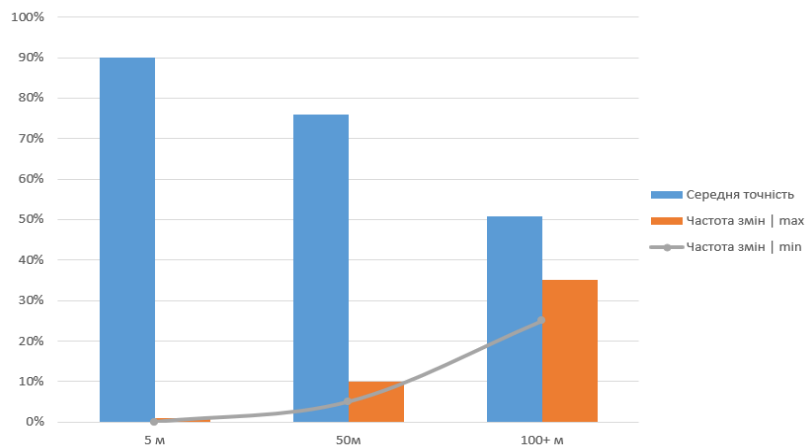


Рис. 9 Показники імовірності належності знайденого об'єкта до певного класу

На основі аналізу 10000 кадрів відеопотоку було визначено середній рівень точності класифікації та встановлено максимальні й мінімальні значення похибки. Результати чітко показали пряму залежність між збільшенням відстані та зниженням точності: зі зростанням віддалі впевненість моделі в класифікації поступово зменшується, а кількість помилок розпізнавання зростає.

Експериментальне тестування підтвердило високу ефективність запропонованого методу розпізнавання об'єктів у реальному часі для БПЛА в задачах розслідувань і пошукових робіт. Система продемонструвала стабільну роботу в умовах змінного освітлення, складного фону, великої кількості об'єктів та динамічних сцен. Модель виявила стійкість до хибних спрацювань і здатність функціонувати в безперервному режимі без деградації швидкодії, ефективно обробляючи одночасно декілька об'єктів. Точність розпізнавання залишається високою, хоча і демонструє залежність від відстані.

Швидкодія системи зберігається навіть при зростанні кількості об'єктів, що критично важливо для її використання в реальному часі на обмежених обчислювальних ресурсах. Надійне виявлення транспортних засобів, людей та військової техніки в різноманітних умовах свідчить про універсальність обраного підходу. Враховуючи отримані результати, розроблений метод можна успішно застосовувати у сферах, що потребують швидкої візуальної аналітики: аеророзвідка, оперативне реагування під час надзвичайних ситуацій, моніторинг дорожньої інфраструктури, забезпечення безпеки масових заходів, виявлення порушень на промислових об'єктах тощо. Гнучкість архітектури системи дозволяє адаптувати її до інших задач комп'ютерного зору з мінімальними зусиллями.

Висновки

У рамках розробки методу було підтверджено, що він значно підвищує ефективність пошукових робіт та розслідувань, надаючи передові рішення для розпізнавання об'єктів у реальному часі за допомогою БПЛА. Це досягнуто завдяки глибокому опрацюванню технологій комп'ютерного

зору, що лежать в основі виявлення об'єктів для цих завдань, з особливим акцентом на практичну реалізацію. У процесі розробки було ретельно проаналізовано актуальність завдання, досліджено наявні підходи, створено унікальний метод розпізнавання об'єктів у реальному часі та розроблено алгоритм виявлення на відеопотоці. Система пройшла всебічне тестування та експериментальні дослідження, підтвердивши свою ефективність.

Результати тестування свідчать про високу ефективність запропонованого підходу. Середній рівень точності розпізнавання складав 90% на відстані до 5 м, 60% на 50 м, 50% на понад 100 м. Коливання точності не перевищували 1% на 5 м, 10% на 50 м і 35% при 100+ м, що свідчить про стабільну роботу в умовах різних відстаней.

Швидкодія системи в процесі обробки оцінювалась за такими метриками, що складав середній час препроцесингу кадру складав $5,5 \pm 4,3$ мс, середній час інференсу складав $3,8 \pm 1,4$ мс, а середній час обробки одного об'єкта 3 ± 1 мс.

Аналіз частоти кадрів демонструє максимальний FPS в 41 кадр, а мінімальний 27, також при виявленні 1 об'єкту 28-31 кадр, при 2-3 об'єктах 25-28 кадрів, у випадках більше 3 кадри зменшувалися до 18-20, а при відсутності відповідно піднімалися до 33-40 кадрів.

Розроблений метод є сучасним технічним рішенням, що поєднує інтелектуальний аналіз, простоту використання, високу швидкодію та технологічну відкритість. Метод є гнучким для подальшої адаптації, що дозволяє інтегрувати його в криміналістичні операції, системи автоматизованих спостережень або пошуково-рятувальні операції. У перспективі можливе подальше удосконалення алгоритмів, розширення набору розпізнаваних класів, оптимізація, а також створення веб-інтерфейсу чи повноцінного застосунку, що дозволить масштабувати розроблену систему для промислового чи польового розгортання.

Література

1. Навчання з підкріпленням для систем спостереження БПЛА [Електронний ресурс] / Коя Кікучі // Матеріали щорічної конференції JSME з робототехніки та мехатроніки (Robomec). – 2019. – Т. 2019. – С. 1P1–M06.
2. . Ван Ш. Дослідження алгоритмів серії YOLO: порівняння та аналіз моделей виявлення об'єктів для застосування в реальному часі на БПЛА / Шуо Ван // Journal of Physics: Conference Series. – 2021. – Т. 1948, № 1. – С. 012021.
3. . Лю С. Дослідження вдосконалення алгоритму YOLOv3 / Сінчао Лю, Хайюн Ган, Їн Ян // Journal of Physics: Conference Series. – 2021. – Т. 1884, № 1. – С. 012031.
4. Урахування швидкості та індивідуалізоване розподілення завдань і ресурсів у мобільних периферійних обчисленнях / Далі Чжу // IEEE Communications Letters. – 2021. – Т. 25, № 8. – С. 2683–2687.
5. Флоріан Н. Система виявлення втом в реальному часі з використанням Nvidia Jetson Nano та OpenCV / Ніколае Флоріан, Думітру Попеску, Андрей Хоссу // Procedia Computer Science. – 2024. – Т. 242. – С. 536–543.
6. Редді Дірісала С. Порівняльне дослідження алгоритмів виявлення об'єктів: YOLOv4, SSD та RCNN на основі точності та швидкості / Сауріш Редді Дірісала, Діліпкumar Падідем // International Journal of Science and Research (IJSR). – 2023. – Т. 12, № 10. – С. 1560–1565.
7. Підвищення надійності виявлення об'єктів за несприятливих погодних умов / Шаїк Якоб [та ін.] // Procedia Computer Science. – 2025. – Т. 252. – С. 1014–1024.
8. Нізамія Н. Система виявлення та фіксації пошкоджень доріг з використанням YOLOv7: підвищення участі громадськості в обслуговуванні інфраструктури / Нізамія Нізамія, Мухамад Брілліант // International Journal of Engineering Continuity. – 2024. – Т. 3, № 2. – С. 69–85.
9. Цифрово прозорий інтерфейс із використанням відстеження очей / Джай Пракаш // Procedia Computer Science. – 2016. – Т. 84. – С. 57–64.
10. PyTorch distributed / Шен Лі // Proceedings of the VLDB Endowment. – 2020. – Т. 13, № 12. – С. 3005–3018.
11. Рахім А. Підхід на основі Ultralytics YOLOv8 для виявлення доріг у сніжних умовах Арктичного регіону Норвегії / Акса Рахім, Фушін Юань, Джавад Барабаді // Computers, Materials & Continua. – 2025. – С. 1–10.
12. Карааслан Ю., Гелішкен У., Куюмджу Б. OpenCV / Юнус Карааслан, Угур Гелішкен, Бірол Куюмджу. – [Місце видання не зазначено] : Level, 2022. – 320 с.
13. Неллі Ф. Бібліотека NumPy / Фабіо Неллі // Python Data Analytics. – Берклі, Каліфорнія, 2018. – С. 49–85.
14. Чанг Е. Свіжа PyQt5: посібник для початківців / Едвард Чанг. Independently Published, 2020.
15. PyGetWindow. PyGetWindow: простий у використанні модуль для роботи з вікнами [Електронний ресурс] / PyGetWindow. – Режим доступу: <https://pypi.org/project/PyGetWindow/>.
16. pywin32. pywin32: Розширення Python для Windows [Електронний ресурс] / pywin32. – Режим доступу: <https://pypi.org/project/pywin32/>.

17. Kaggle. Набір даних виявлення автомобілів з дрону з вигляду згори [Електронний ресурс] / Kaggle. – Режим доступу: <https://www.kaggle.com/datasets/glebkuzntsov/top-view-drone-car-detection-dataset-12000-images>.
18. Kaggle. Набір даних YOLO для виявлення осіб і сигнального одягу [Електронний ресурс] / Kaggle. – Режим доступу: <https://www.kaggle.com/datasets/tudorhirtopanu/yolo-highvis-and-person-detection-dataset>
19. Kaggle. Набір даних для виявлення людей [Електронний ресурс] / Kaggle. – Режим доступу: <https://www.kaggle.com/datasets/constantinwerner/human-detection-dataset>.
20. Kaggle. Набір даних кадрів JAAD [Електронний ресурс] / Kaggle. – Режим доступу: <https://www.kaggle.com/datasets/charvik/jaad-frames-dataset-10-v1-yolo-format>.
21. Kaggle. Набір даних військових об'єктів [Електронний ресурс] / Kaggle. – Режим доступу: <https://www.kaggle.com/datasets/rawsi18/military-assets-dataset-12-classes-yolo8-format>.

References

1. Reinforcement Learning for UAV Surveillance Systems / Koya KIKUCHI // The Proceedings of JSME annual Conference on Robotics and Mechatronics (Robomec). – 2019. – Vol. 2019. – P. 1P1–M06.
2. Wang S. Research Towards Yolo-Series Algorithms: Comparison and Analysis of Object Detection Models for Real-Time UAV Applications / Shuo Wang // Journal of Physics: Conference Series. – 2021. – Vol. 1948, no. 1. – P. 012021.
3. Liu X. Study on Improvement of YOLOv3 Algorithm / Xinchao Liu, Haiyun Gan, Ying Yan // Journal of Physics: Conference Series. – 2021. – Vol. 1884, no. 1. – P. 012031.
4. Speed-Aware and Customized Task Offloading and Resource Allocation in Mobile Edge Computing / Dali Zhu // IEEE Communications Letters. – 2021. – Vol. 25, no. 8. – P. 2683–2687.
5. Florian N. Real-time Tiredness Detection System using Nvidia Jetson Nano and OpenCV / Nicolae Florian, Dumitru Popescu, Andrei Hossu // Procedia Computer Science. – 2024. – Vol. 242. – P. 536–543.
6. Reddy Dirisala S. Comparative Study of the Different Object Detection Algorithms: YOLOv4, SSD, and RCNN based on Accuracy and Speed / Saurish Reddy Dirisala, Dileepkumar Padidem // International Journal of Science and Research (IJSR). – 2023. – Vol. 12, no. 10. – P. 1560–1565.
7. Enhancing Object Detection Robustness In Adverse Weather Conditions / Shaik Yacoob [et al.] // Procedia Computer Science. – 2025. – Vol. 252. – P. 1014–1024.
8. Nizamiyati N. Road Damage Detection and Reporting System Using YOLOv7: Enhancing Public Participation in Infrastructure Maintenance / Nizamiyati Nizamiyati, Muhamad Brilliant // International Journal of Engineering Continuity. – 2024. – Vol. 3, no. 2. – P. 69–85.
9. Digitally Transparent Interface Using Eye Tracking / Jai Prakash // Procedia Computer Science. – 2016. – Vol. 84. – P. 57–64.
10. PyTorch distributed / Shen Li // Proceedings of the VLDB Endowment. – 2020. – Vol. 13, no. 12. – P. 3005–3018.
11. Rahim A. An Ultralytics YOLOv8-Based Approach for Road Detection in Snowy Environments in the Arctic Region of Norway / Aqsa Rahim, Fuqing Yuan, Javad Barabady // Computers, Materials & Continua. – 2025. – P. 1–10.
12. Yunus Karaaslan, Ugur Gelisken, Birol Kuyumcu. OpenCv / Yunus Karaaslan, Ugur Gelisken, Birol Kuyumcu. – [S. l.] : Level, 2022. – 320 p.
13. Nelli F. The NumPy Library / Fabio Nelli // Python Data Analytics. – Berkeley, CA, 2018. – P. 49–85.
14. Chang E. Fresher PyQt5: A Beginner's Guide to PyQt5 / Edward Chang. – [S. l.] : Independently Published, 2020.
15. PyGetWindow. PyGetWindow: A simple and easy-to-use module for getting information about and working with windows [Electronic resource] / PyGetWindow. – Mode of access: <https://pypi.org/project/PyGetWindow/>.
16. pywin32. pywin32: Python for Windows Extensions [Electronic resource] / pywin32. – Mode of access: <https://pypi.org/project/pywin32/>.
17. Kaggle. Top View Drone Car Detection Dataset [Electronic resource] / Kaggle. – Mode of access: <https://www.kaggle.com/datasets/glebkuzntsov/top-view-drone-car-detection-dataset-12000-images>.
18. Kaggle. YOLO High-Vis and Person Detection Dataset [Electronic resource] / Kaggle. – Mode of access: <https://www.kaggle.com/datasets/tudorhirtopanu/yolo-highvis-and-person-detection-dataset>
19. Kaggle. Human Detection Dataset [Electronic resource] / Kaggle. – Mode of access: <https://www.kaggle.com/datasets/constantinwerner/human-detection-dataset>.
20. Kaggle. JAAD Frames Dataset [Electronic resource] / Kaggle. – Mode of access: <https://www.kaggle.com/datasets/charvik/jaad-frames-dataset-10-v1-yolo-format>
21. Kaggle. Military Assets Dataset [Electronic resource] / Kaggle. – Mode of access: <https://www.kaggle.com/datasets/rawsi18/military-assets-dataset-12-classes-yolo8-format>.