
СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

УДК 681.516+004.932.2

С.П. РОБОТЬКО, А.М. ТОПАЛОВ

БАГАТОРІВНЕВА АРХІТЕКТУРА СИСТЕМИ АВТОМАТИЧНОГО КЕРУВАННЯ БПЛА ДЛЯ ЗДІЙСНЕННЯ ПОШУКОВИХ МІСІЙ ЗА ВІДЕОАНАЛІЗОМ ТА МЕТАЛОДЕТЕКЦІЄЮ

Національний університет кораблебудування імені адмірала Макарова

Анотація. У статті представлено багаторівневу систему автоматичного керування місіями безпілотного літального апарата у задачах виявлення підозрілих предметів. Запропонована архітектура поєднує edge-ground-cloud обробку даних з бортової відеокамери та металодетектора, а також використання візуально-мовних моделей (ChatGPT-4.1 Vision, Gemini 2.5 Flash) для семантичної верифікації підозрілих предметів. На наземній станції виконується первинна детекція небезпечних предметів за допомогою YOLOv8 та аналіз сигналу металодетектора, після чого кадри з проміжною впевненістю надсилаються у хмару для додаткової перевірки VLM. На основі об'єднаної оцінки формується рішення про наявність небезпечного предмета, яке через MAVLink автоматично коригує місію БПЛА завдяки чому дрон переводиться з режиму AUTO в GUIDED, повертається за GPS-координатами підозри, виконує додаткове обстеження та після цього відновлює маршрут із збереженої точки. Експериментальні випробування на полігоні з макетами небезпечних предметів показали, що комбінування YOLOv8, металодетектора та VLM дозволяє досягти підвищення точності до $\approx 95,7\%$ і збереженні роботи в режимі, близькому до реального часу (ефективно 5 кадрів/с). Наукова новизна роботи полягає в реалізації замкненого циклу «виявлення – семантична перевірка – автоматичне коригування місії» для БПЛА, що поєднує мультимодальне злиття даних і хмарні AI-моделі та знижує навантаження на оператора.

Ключові слова: безпілотні літальні апарати (БПЛА), комп'ютерний зір, багаторівневий аналіз відеозображень, металодетектор, візуально-мовні моделі, автоматизована система керування, зворотний зв'язок, багаторівнева архітектура, MAVLink.

Abstract: The article presents a multi-level automatic mission control system for an unmanned aerial vehicle designed to detect hazardous items in tasks involving the identification of suspicious objects. The proposed architecture combines edge-ground-cloud data processing from the onboard video camera and metal detector, as well as the use of vision-language models (ChatGPT-4.1 Vision, Gemini 2.5 Flash) for semantic verification of suspected objects. At the ground station, initial detection of hazardous items is performed using YOLOv8 and metal-detector signal analysis. Frames with intermediate confidence are then sent to the cloud for additional verification by VLMs. Based on the combined assessment, a decision is generated regarding the presence of a hazardous item, which automatically adjusts the UAV mission via MAVLink: the drone is switched from AUTO to GUIDED mode, returns to the GPS coordinates of the suspicion, performs additional inspection, and then resumes the mission from the saved waypoint.

Experimental field tests with mock-ups of hazardous items demonstrated that combining YOLOv8, the metal detector, and VLMs makes it possible to achieve increasing precision to approximately 95.7% and maintaining near-real-time performance (effective 5 fps). The scientific novelty of the work lies in implementing a closed loop of “detection – semantic verification – automatic mission correction” for UAVs, which integrates multimodal data fusion with cloud-based AI models and reduces operator workload.

Keywords: unmanned aerial vehicles (UAVs), computer vision, multi-level video-image analysis, metal detector, vision-language models, automatic control system, feedback, multi-level architecture, MAVLink.

DOI: 10.31649/1681-7893-2025-50-2-114-124

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

ВСТУП

Проблема дистанційного виявлення небезпечних предметів є надзвичайно актуальною. Традиційні методи виявлення вимагають значних людських ресурсів, тому все більшої уваги набувають безпілотні літальні апарати (БПЛА) як засоби швидкого дистанційного обстеження великих площ. Однак використання лише одного типу датчика на борту дрона не гарантує надійного виявлення, тому що класичні металодетектори ефективні для масивних цілей, але дрібні предмети часто залишаються поза їх чутливістю або дають неоднозначні сигнали за нестабільної висоти чи траєкторії польоту БПЛА. З іншого боку, комп'ютерний зір з UAV довів ефективність у візуальному виявленні небезпечних предметів, але стикається з такими викликами як мінімальні розміри небезпечних предметів, камуфляж, складний фон (рослинність, тінь), змінні умови освітлення знижують надійність чисто оптичної детекції [5,6]. Жоден з відомих підходів не дає 100% гарантії – в оглядах підкреслюється, що наразі не існує універсального методу з прийнятною надійністю для виявлення небезпечних предметів з БПЛА [7]. Тому актуальним є поєднання кількох сенсорів та рівнів обробки [8], аби компенсувати обмеження кожного з підходів. Зокрема, рекомендації сучасних досліджень вказують на доцільність інтеграції оптичних, тепловізійних, магнітометричних (металодетектор) та інших сенсорів для підвищення ймовірності виявлення небезпечних предметів [9]. Крім того, типові системи керування БПЛА [10] досі базуються на ручному управлінні або жорстко запрограмованих сценаріях, що обмежує їхню гнучкість та автономність. Усе це визначає потребу в нових архітектурах: багаторівневих системах, які розподіляють завдання між бортом, наземною станцією та хмарою, адаптивно поєднуючи різні модальності даних і автоматично втручаючись у місію при виявленні загрози.

Метою даного дослідження є розробити інноваційну багаторівневу систему керування місією БПЛА для виявлення небезпечних предметів у режимі, близькому до реального часу, що поєднує комп'ютерний зір і металодетекцію. Така система має автоматично реагувати на виявлені підозрілі предмети: повторно обстежувати їхню зону за GPS-координатами та коригувати маршрут дрона без участі оператора. Це дозволить знизити ризики для людей та підвищити достовірність пошуку небезпечних предметів.

1. ОГЛЯД ЛІТЕРАТУРИ ТА АНАЛОГІВ ДОСЛІДЖЕНЬ

Сучасні детектори предметів сімейства YOLO зарекомендували себе високою швидкістю та точністю для різних задач розпізнавання. Зокрема, модель YOLOv8 була успішно застосована для пошуку наземних небезпечних предметів, забезпечивши виявлення у реальному часі з високою повнотою [1]. Перевагою YOLOv8 є компактність і швидкодія: найменша модель (nano) після конвертації до формату ONNX має ~3 МБ і здатна працювати навіть на смартфоні зі швидкістю ~2 кадри/с, що свідчить про можливість розгортання на малопотужних пристроях (на кшталт Raspberry Pi на борту). Утім, викликом є помилкові спрацьовування: дослідження показали, що YOLOv8 може давати до ~33–35% хибнопозитивних визначень фону як небезпечних предметів. В контексті підозрілих предметів це допустимо – краще виявити зайве, ніж пропустити загрозу. Дійсно, автори роботи наголошують, що висока чутливість системи на базі YOLO є виправданою, тому що хибні спрацьовування можна швидко відсіяти оператором, зате ризик пропустити справжню загрозу мінімізовано. Наша система розвиває цю ідею, так як замість людини для перевірки первинних спрацювань використовуються розумні хмарні моделі, що автоматично аналізують підозрілі зображення.

Для підвищення надійності виявлення небезпечних предметів перспективним є об'єднання даних різної природи. Наприклад, у роботі Станкевича та Саприкіна [3] запропоновано об'єднувати оптичні зображення з сигналами магнітометра на борту дрона. Оптичні дані аналізувалися згортковою нейромережею (CNN), а магнітометричні – методом статистичного виявлення аномалій. На рівні прийняття рішень результати двох каналів інтегрувались за Bayesian-правилом. Комбінація видимого оптичного каналу та магнітометра суттєво підвищила надійність виявлення наземних небезпечних предметів. В інших роботах для злиття даних з сенсорів випробовувалися також методи теорії довіри Демпстера-Шефера та нечіткої логіки, що дозволяло скоротити число хибних тривог при одночасному використанні металопрошукача і радара або камери. Отже, залучення додаткового сенсора (металодетектора) поряд з комп'ютерним зором обґрунтовано попередніми дослідженнями як шлях до підвищення достовірності виявлення небезпечних предметів.

Швидкий прогрес мультимодальних великих мовних моделей (Vision-Language Models, VLM) відкрив нові можливості для автономних роботів, у тому числі БПЛА [2]. На відміну від традиційних алгоритмів комп'ютерного зору, які видають лише клас предмета чи координати, VLM здатні семантично інтерпретувати зображення у контексті. Це означає, що модель типу GPT-4 з модулем

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

комп'ютерного зору може не тільки впізнати предмет, але й пояснити, що саме зображено, зробити висновки про ситуацію тощо. Як показано в роботі Крупаша та співавт., впровадження VLM у контур керування дроном дозволяє апарату приймати контекстно обґрунтовані рішення та навіть надавати користувачу пояснення своїх дій у природній мові. Зокрема, їхня система з GPT-4.1 Vision навчилася генерувати навігаційні команди для дрона та супроводжувати їх поясненнями і рівнем впевненості, що підвищує довіру оператора та прозорість роботи автономного БПЛА. Таким чином, VLM можуть виступати в ролі "розумного аналітика" на борту або в хмарі, доповнюючи класичний комп'ютерний зір.

Втім, варто зазначити, що універсальні мультимодальні моделі поки поступаються вузькоспеціалізованим алгоритмам у точності та швидкодії на конкретних завданнях. Як показує бенчмарк, GPT-4 Vision значно програє YOLOv8 у задачі детекції предметів: середня точність виявлення (mAP@0.5) для GPT-4 становить лише ~0.02, тоді як у компактній моделі YOLOv8n – ~0.20. Крім того, обробка зображень GPT-4 є на порядки повільнішою (латентність ~5,15 с на зображення) порівняно з ~0,36 с у YOLOv8n. Таким чином, безпосередньо використовувати VLM для суцільного моніторингу відеопотоку нерационально. Наша система поєднує сильні сторони обох підходів: швидкий огляд сцени за допомогою YOLOv8 на борту, а лише для фрагментів, що викликають підозру, – глибокий аналіз VLM-моделями в хмарі. Така дворівнева схема дозволяє досягти як оперативності, так і змістовності аналізу. Крім того, використання двох незалежних VLM (ChatGPT-4.1 Vision від OpenAI та Gemini-2.5 Flash від Google) дає змогу перехресно перевіряти результат і зменшити ймовірність помилки окремої моделі. Обидві ці моделі підтримують введення зображень та генерують текстовий висновок, що робить їх придатними для нашої задачі семантичного розпізнавання знайдених предметів.

Отже, аналіз літератури показує доцільність багаторівневого підходу: поєднання edge-аналізу (оперативний комп'ютерний зір) із хмарним інтелектуальним аналізом та сенсорним підтвердженням дозволяє підвищити надійність і зрозумілість виявлення небезпечних предметів. Наступні розділи детально описують постановку задачі, архітектуру системи та реалізацію запропонованого підходу.

2. ПОСТАНОВКА ЗАДАЧІ ТА КЛЮЧОВІ АСПЕКТИ

Постановка проблеми. Необхідно забезпечити БПЛА можливість автономно виявляти небезпечні предмети під час патрулювання території та адаптивно коригувати свій політ для детальнішого обстеження підозрілих зон. Система має працювати в реальному часі, обробляючи відеопотік з бортової камери, і приймати рішення про зміну маршруту без участі людини-оператора. Ключова вимога – висока ймовірність виявлення (не пропустити загрозу) при прийнятному рівні хибних тривог. Останнє означає, що система може помилятися у бік зайвого спрацьовування, але кожне спрацьовування повинно бути перевірене додатково, аби відсіяти помилки до прийняття остаточного рішення.

Запропонований підхід передбачає багаторівневу архітектуру, яка об'єднує три послідовні рівні аналізу даних. На бортовому рівні edge працює модуль комп'ютерного зору на базі YOLOv8, розгорнутий на компактному комп'ютері Raspberry Pi, підключеному до автопілота Pixhawk під керуванням ArduPilot. Цей модуль у реальному часі аналізує відеопотік з камери з частотою тридцять кадрів за секунду, але для зменшення навантаження відбирає лише близько п'яти кадрів за секунду для подальшої обробки. YOLOv8 виявляє потенційно небезпечні предмети, наприклад контури підозрілих предметів чи снарядів, на кожному зображенні та для кожної детекції формує область інтересу з оцінкою впевненості моделі. Якщо довірча ймовірність перевищує заданий поріг, предмет вважається підозрілим і передається на наступний етап. Другий рівень реалізовано у хмарному середовищі, де для кожної підозрілої області інтересу формується асинхронний запит до серверів зі зображенням та коротким текстовим описом ситуації у вигляді prompt. Застосовуються дві незалежні візуально-мовні моделі, а саме ChatGPT 4.1 Vision та Gemini 2.5 Flash. Вони аналізують фрагмент зображення і повертають текстовий висновок із описом предмета та припущенням щодо його природи, наприклад чи нагадує він пластиковий небезпечний пристрій типу ПФМ-1 або є звичайним каменем. Якщо хоча б одна з моделей впевнено вказує на небезпечний предмет, система сприймає це як підтвердження загрози. Використання двох різних моделей підвищує надійність, оскільки мультимодальні системи іноді можуть помилятися або генерувати хибні уявлення, а перехресна перевірка дає змогу підсилити впевненість у рішенні. Враховуючи значну тривалість хмарного аналізу, що в середньому становить кілька секунд на одне зображення для GPT 4, частоту запитів обмежено тими ж п'ятьма кадрами за секунду, а запити до ChatGPT і Gemini виконуються паралельно. Для такої паралельної обробки потрібен доступ до програмних інтерфейсів моделей з рівнем сервісу не нижче Tier 3, що дає змогу виконувати кілька звернень одночасно без перевищення лімітів. Третій рівень пов'язаний з роботою додаткового сенсора, яким є малогабаритний металодетектор або магнітометр, встановлений на БПЛА і здатний виявляти масивні металеві предмети під апаратом. Під час звичайного пошукового польоту цей сенсор працює у

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

фоновому режимі, проте через досить велику висоту польоту, зазвичай близько п'яти-десяти метрів, його чутливості може не вистачати для дрібних небезпечних предметів. Найбільший ефект досягається на етапі детального обстеження, коли апарат знижується і пролітає безпосередньо над підозрілим предметом. У розробленій системі металодетектор активно використовується саме в момент повторного обльоту цілі, коли БПЛА опускається до безпечної малої висоти близько двох метрів над місцем, де було зафіксовано підозру, і зчитує показання магнітометра. Виявлення локальної аномалії магнітного поля або сигналу металодетектора виступає додатковим підтвердженням наявності металу, що є типовою ознакою небезпечних предметів. Об'єднання оптичної інформації з відеоканалу та магнітних ознак суттєво підвищує надійність виявлення, що узгоджується з результатами попередніх досліджень і дає змогу зменшити кількість як хибних пропусків, так і зайвих спрацьовувань.

На основі даних трьох рівнів система повинна вирішити, чи вважати знайдений предмет вибухонебезпечним і, відповідно, чи варто змінити маршрут для його перевірки. Формально, рішення можна подати як функцію:

$$P_{\text{final}} = f(s_Y, s_C, s_G, s_M), \quad (1)$$

де s_Y, s_C, s_G, s_M – ознаки виявлення на рівні YOLO, ChatGPT, Gemini та металодетектора відповідно. Тут s_i можуть бути булевими індикаторами (1 – загроза виявлена, 0 – ні) або ж певними балами впевненості/ймовірності. Функція f може бути визначена за різними правилами: Логічна схема (правила OR/AND): Найчутливіший підхід – правило логічного OR, при якому загроза підтверджується, якщо *хоча б один* з каналів дав сигнал (наприклад, або YOLO, або VLM, або металодетектор). Така схема гарантує мінімум пропусків, але може давати надлишок помилкових спрацьовувань. Більш консервативний підхід – AND вимагає, щоб кілька незалежних ознак співпали. Наприклад, можна вирішити виконати повторну перевірку лише якщо *одночасно* YOLO детектувала предмет і хоча б одна VLM-модель класифікувала його як підозрілий (YOLO AND VLM), або якщо VLM дала позитивний висновок і спрацював металодетектор (VLM AND Metal). Логічні правила надають інтуїтивно зрозумілу схему ескалації: система може бути налаштована або на максимальну чутливість (OR всіх ознак), або на максимальну специфічність (вимога підтвердження кількох каналів). Альтернативно, можна обчислювати інтегральний показник як лінійну комбінацію сигналів:

$$P_{\text{final}} = \lambda_Y s_Y + \lambda_C s_C + \lambda_G s_G + \lambda_M s_M, \quad (2)$$

де λ_{i_i} – вагові коефіцієнти каналу. Наприклад, якщо металодетектор менш чутливий, але дуже надійний, йому можна призначити високий коефіцієнт λ_M , а вагу λ_Y для YOLO навпаки трохи зменшити, враховуючи його схильність до хибних спрацьовувань. Підсумковий P_{final} порівнюється з порогом. Такий підхід фактично реалізує Bayesian-правило або методи теорії ймовірностей для злиття даних, подібно до того, як це зроблено в роботі, де оптична і магнітна ознаки комбінуються за допомогою апостеріорної ймовірності.

У нашій реалізації використано комбінацію підходів. Спершу застосовується логічне правило ескалації: якщо $s_Y=1$ (YOLO щось знайшла), то запускається хмарний аналіз s_C, s_G . Остаточне рішення щодо маневру приймається за правилом (YOLO AND (ChatGPT OR Gemini)), тобто потрібне підтвердження хоча б однією VLM-моделлю. При цьому спрацьовування металодетектора на етапі повторного обльоту автоматично означає підтвердження загрози незалежно від інших каналів (сенсорний сигнал важить найбільше). Така схема гарантує, що одиничний результат YOLO (який може бути хибним) не призведе до зриву місії, якщо обидва розумні аналізатори не побачили нічого небезпечного. З іншого боку, якщо VLM «побачили» підозрілий предмет на зображенні, система радше перестрахується і виконає перевірку. Формальні вагові коефіцієнти можна вважати λ_Y помірним, $\lambda_{C,G}$ високими, коли кожному з VLM присвоюється велика вага, λ_M найбільшим. У результаті багаторівнева логіка налаштована на ескалацію, де будь-який предмет, помічений комп'ютерним зором, проходить через все детальніші рівні аналізу, і лише якщо *жоден* з рівнів не підтвердив загрозу – БПЛА продовжує місію без змін.

Ключові аспекти реалізації системи полягають у комбінуванні низки технічних рішень, що забезпечують роботу повного циклу – від отримання сенсорних даних до автоматичного коригування місії БПЛА. Основою є edge-ground-cloud архітектура з використанням реального автопілота. Безпілотник оснащено автопілотом Pixhawk (ArduPilot) та одноплатним комп'ютером Raspberry Pi 4, які взаємодіють через протокол MAVLink. Raspberry Pi виконує функції бортового комп'ютера під Linux і реалізує спеціалізований програмний стек, тоді як автопілот відповідає за стабілізацію та виконання маршруту.

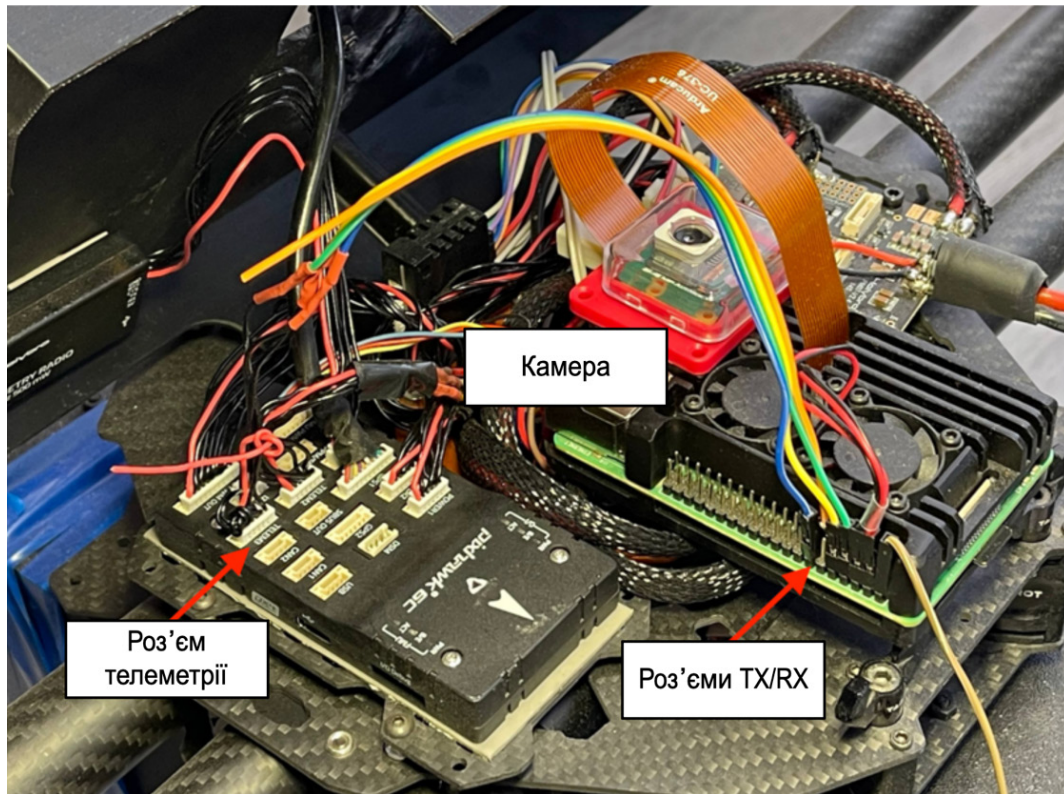


Рисунок 1 – Підключення політного контролера до Raspberry Pi

Саме Raspberry Pi за потреби передає автопілоту команди зміни режиму польоту та корекції координат. Саме підключення політного контролера до одноплатного комп'ютера налаштовано через роз'єми TELEM3, що відповідає UART/USART до TX/RX (Receiver/Transmitter) на Raspberry Pi, як зображено на рисунку 1. Моніторинг і налаштування здійснюються через наземну станцію QGroundControl, підключену по радіотелеметрії. Така структура дозволяє впровадити розподілення обчислювальних задач коли попередня обробка даних виконується на борту, контроль місії – на землі, а ресурсоемні обчислення (візуально-мовні моделі) – у хмарі.

Камера на борту дрона формує відео 1080p із частотою 30 кадрів на секунду, але для забезпечення роботи в реальному часі аналізується лише кожен шостий кадр. Ефективна частота близько 5 кадрів на секунду забезпечує водночас достатню оглядовість при русі та прийнятне навантаження на обчислювальні ресурси. Обраний кадр масштабується до цільового розміру (наприклад, 640×480) і подається на вхід локальної моделі YOLOv8, яка здатна обробляти один кадр за 0,2–0,5 секунди на малопотужних пристроях. Значна частина зображень після локального аналізу відкидається, оскільки не містить потенційно цікавих областей. У хмару надсилаються лише кадри, у яких YOLO визначила регіон інтересу, та лише фрагменти цих кадрів, що потребують додаткової семантичної перевірки. Це принципово важливо, адже моделі на кшталт ChatGPT-4 Vision фізично не здатні стабільно опрацьовувати 5 запитів на секунду, оскільки кожен виклик триває кілька секунд. Обчислення в хмарі відбуваються паралельно з локальним аналізом наступних кадрів, тому цикл роботи системи не зупиняється. На Raspberry Pi вся логіка реалізована мовою Python 3.9 із використанням PyTorch або ONNX Runtime для виконання YOLOv8, а також HTTP-клієнтів для надсилання запитів у хмару (OpenAI та Google Cloud). Підсистема побудована з активним застосуванням багатопотоковості. Окремі потоки відповідають за захоплення кадрів із камери та локальну детекцію, за оброблення хмарних запитів без блокування основного циклу, за взаємодію з автопілотом через MAVLink, а також за опитування металодетектора. Обмін між потоками організований через черги подій, коли YOLO знаходить ROI, відповідна задача передається до потоку хмарних викликів, після чого відбувається отримання відповіді від моделі та аналіз ключових слів або оцінки, на основі яких формується рішення. Потік керування польотом безперервно оцінює об'єднаний стан усіх сенсорів та VLM і за необхідності ініціює зміну

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

режиму польоту. Потоки працюють паралельно, тому дрон продовжує рух, навіть якщо хмарна модель все ще обробляє попередній кадр. Уведено обмеження на частоту хмарних запитів, щоб не перевищувати квоти API та не перенавантажувати канал зв'язку.

Оскільки система використовує дві незалежні візуально-мовні моделі (GPT-4.1 Vision та Gemini 2.5 Flash), її налаштовано з урахуванням комерційних квот та політик провайдерів. Розгортання орієнтоване на використання тарифних планів, що дозволяють підвищену паралельність запитів. У разі збільшення частоти виявлень входить у дію механіка черги, тобто якщо ліміт запитів перевищено, обробляється лише один із двох VLM, або моделі чергуються між собою (наприклад, непарні спрацьовування надсилаються до ChatGPT, парні – до Gemini). У тестовому режимі відповідні квоти Tier-3 дозволили стабільно виконувати до двох паралельних викликів без затримок, а за необхідності масштабування ця кількість може бути збільшена. Усі елементи системи контейнеризовані за допомогою Docker для забезпечення ізоляції, відтворюваності та можливості швидкого перенесення між пристроями. Створено окремі контейнери для локального edge-модуля з YOLOv8 та обробленням відео, для хмарного агента із ключами та інтерфейсами API, а також для бази даних або журналювання. Така структура дає змогу локалізувати можливі відмови: збій хмарного сервісу не впливає на роботу локальної детекції, а кожен контейнер може масштабуватися або переноситися окремо.

Система також містить розвинуту підсистему журналювання з єдиним часовим простором. Усі записи – від кадрів камери та результатів YOLO до відповідей VLM, показань металодетектора та команд керування – мають уніфіковані UTC-мітки з мілісекундною точністю. Raspberry Pi синхронізується через NTP, а автопілот передає GPS-час через MAVLink, що дає змогу повністю узгодити всі джерела даних. Для кожного кадру обчислюється SHA256-хеш, який зберігається разом із результатами аналізу, що забезпечує контроль цілісності та можливість підтвердити, що саме це зображення викликало спрацьовування. Система підтримує режим відтворення, у якому всі логи можуть бути програні повторно для перевірки рішень або налагодження. Такий механізм є критично важливим для застосувань на кшталт гуманітарного розмінування, де достовірність рішень системи має принципове значення.

У підсумку формується цілісний підхід, у якому дрон у реальному часі проходить через цикл багаторівневого аналізу – від комп'ютерного зору до семантичного AI та фізичного сенсора – і на основі цього автоматично коригує свою місію. Запропонована система забезпечує здатність безпілотної самостійно змінювати маршрут для уточнення підозрілих цілей, гарантуючи безперервність руху та високу точність прийняття рішень..

3. МАТЕРІАЛИ ДОСЛІДЖЕННЯ ТА МЕТОДИ

Апаратна частина експериментального стенда складається з квадрокоптера середнього розміру з польотним контролером Pixhawk 2.4.8 на прошивці ArduCopter 4.3 та бортовим комп'ютером Raspberry Pi 4 Model B з обсягом пам'яті 4 GB. На дроні встановлено камеру Arducam 64MP з роздільною здатністю 64 MP на підвісі, орієнтовану донизу для огляду поверхні землі. В якості імпульсного металодетектора викорисотвується модель Tracker-2 з рамкою 1x1 метр, під'єднаний до Raspberry Pi через шину звуковий модуль. Наземна станція представлена ноутбуком з програмою QGroundControl версії 4.0, який з'єднаний з Pixhawk через радіомодем на частоті 915 МГц. Бортвий Raspberry Pi зв'язаний з автопілотом по інтерфейсу UART на порту Telem 3 з використанням протоколу MAVLink. На Raspberry Pi запущено операційну систему Raspberry Pi OS на основі Linux та середовище Docker. Програмне забезпечення реалізовано мовою Python версії 3.9. Для детектора YOLOv8 застосовано бібліотеку Ultralytics з моделлю yolov8n.pt, попередньо донавченою на спеціалізованому наборі даних із зображеннями небезпечних предметів, який містить близько п'ятисот фотографій типових предметів та їхніх фрагментів, зібраних з відкритих джерел. Модель перетворено у формат ONNX і виконують через ONNX Runtime, що дає змогу підвищити швидкодію на ARM-процесорі. Візуально-мовні моделі викликаються через офіційні програмні інтерфейси. Для OpenAI ChatGPT 4.1 використовується REST API з параметром моделі gpt-4-vision, а для Google Gemini 2.5 Flash застосовується інтерфейс Vertex AI з указанням моделі gemini-2.5 flash. Обидва хмарні сервіси доступні через інтернет. У ході експериментів Raspberry Pi підключено до мережі Wi-Fi наземної станції, яка має вихід у глобальну мережу. Середня затримка мережі становить приблизно від п'ятдесяти до ста мілісекунд, що є незначним порівняно з часом роботи моделей. Усі часові параметри вимірювалися та логувалися для подальшого аналізу продуктивності системи.

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

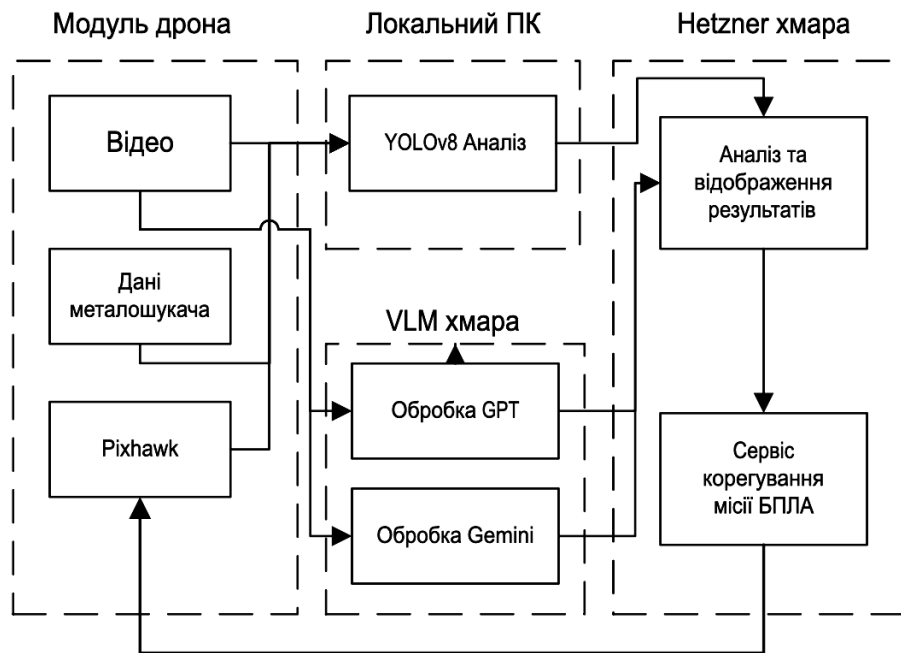


Рисунок 2 – Схема роботи комплексу зі зворотнім зв'язком

Послідовність роботи системи (Рисунок 2) можна описати так. Спочатку оператор у QGroundControl задає маршрут обльоту, наприклад лінійний або сітковий політ над полем. Pixhawk отримує список точок маршруту і переводиться у режим автоматичного польоту, після чого БПЛА злітає і починає виконувати місію. Raspberry Pi, підключений через MAVLink як ведений комп'ютер, отримує потік телеметрії з поточними координатами, висотою та станом режиму і контролює процес виконання місії, яка на цьому етапі проходить у режимі пасивного моніторингу. Потік відео з камери надходить до Raspberry Pi, і кожен N-й кадр, відправляється на наземну станцію, або при наявності LTE зв'язку напряму до хмари. Модель YOLO здійснює пряме поширення та повертає список детекцій із координатами прямокутників, класами та значеннями довіри. Якщо жоден предмет не знайдено або всі мають низьку впевненість, кадр відкидається, і обробка продовжується для наступних кадрів. Коли YOLO виявляє предмет з довірою вище заданого порогу (наприклад понад 50%), запускається процедура ескалації. З кадру вирізається область інтересу з невеликим запасом по краях, формується текстовий контекст з орієнтовними координатами, висотою польоту та описом типу місцевості, після чого ці дані ставляться у чергу для хмарного аналізу. Паралельно місія триває, дрон не зупиняється, і наступні кадри й надалі аналізуються YOLO.

На етапі хмарного аналізу область інтересу надсилається до API моделей ChatGPT 4.1 Vision і Gemini 2.5 Flash, за можливості запити виконуються паралельно. Підказка формулюється англійською мовою та описує контекст того, що дрон зафіксував підозрілий предмет на землі і необхідно визначити, чи є він небезпечним або нешкідливим. Моделі аналізують зображення та повертають текстові відповіді. ChatGPT може описати, що на зображенні видно частково заглиблений у ґрунт зелений предмет, схожий на небезпечний пристрій типу PMN-2, тож є висока ймовірність, що це небезпечний пристрій, або навпаки повідомити, що це заіржавіла бляшанка чи інший безпечний предмет. Gemini формує аналогічний висновок своїми формулюваннями. Після отримання відповідей програма на Raspberry Pi виконує простий лінгвістичний аналіз, шукаючи ключові слова та оцінюючи тон висновку. Якщо модель прямо згадує небезпечний предмет або вживає формулювання на кшталт висока ймовірність небезпечного пристрою, встановлюється сигнал підтвердженої небезпеки, якщо ж у відповіді йдеться про безпечний предмет або сміття, сигнал вважається нульовим. У випадку невпевнених відповідей, коли модель вагається між каменем і небезпечним предметом, система може інтерпретувати це як часткове підтвердження і обирати більш обережну тактику. У виконаних експериментах застосовувалася стратегія, коли позитивний висновок хоча б однієї з двох моделей вважався достатнім для підозри, тому логічний результат трактувався як диз'юнкція сигналів, що відповідає принципу діяти з запасом безпеки.

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

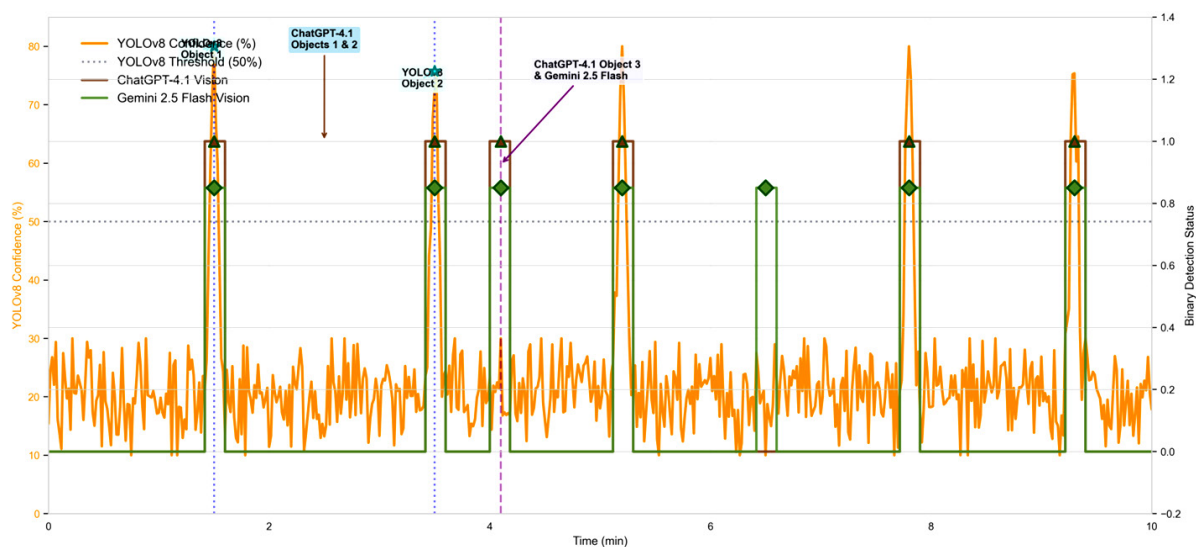


Рисунок 3 – Графік спрацювання систем комплексу

Архітектура системи зображена на рисунку 3 і описує автоматизований пошук небезпечних предметів у багаторівневій схемі edge-ground-cloud. БПЛА патрулює територію в режимі AUTO за наперед заданим маршрутом, який на схемі позначено синім кольором. На борту постійно виконується запис та передача відео, після чого модель YOLOv8 виявляє потенційні загрози і виділяє їх прямокутниками. Паралельно кожен 6-й кадр передається через інтернет у хмару, де візуально-мовна модель ChatGPT та Gemini виконує семантичний аналіз та класифікацію предмета як небезпечного чи безпечного. Якщо за результатами аналізу загроза підтверджується відповідно до заданої логіки, бортовий комп'ютер переводить дрон у режим GUIDED і формує команду на повторний обліт. БПЛА прямує до координат виявленого предмета, знижуючи висоту для детальнішого огляду. Під час цього повторного проходу активується металодетектор, і якщо він реєструє наявність металу, це додатково підтверджує наявність небезпечного предмета. Після завершення перевірки дрон повертається до виконання основної місії в режимі AUTO. На всіх етапах відбувається запис логів телеметрії, результатів аналізу зображень і показників сенсорів. На основі результатів хмарного аналізу система приймає рішення щодо маневру. На цьому кроці відомо, що хоча б один з рівнів (металодетектор, Yolo, ChatGPT4.1 або Gemini 2.5 Flash) спрацював. Якщо YOLO зафіксував предмет і принаймні одна з VLM підтвердила можливу небезпеку, система розцінює це як підставу для повторної перевірки. При одночасному спрацюванні мінімум 3-х рівнів детекції, дрон продовжує місію без змін, зафіксувавши наявність підозрілого предмета, що вимагає уваги оператора. Така багаторівнева фільтрація суттєво зменшує кількість переривань польоту, що особливо помітно в умовах складного рельєфу, де YOLO може давати кілька хибних спрацювань за годину, тоді як візуально-мовні моделі відсіюють більшість із них як очевидно безпечні.

Коли система вирішує виконати повторний огляд, Raspberry Pi надсилає через MAVLink команду на перехід автопілота в режим GUIDED. З цього моменту ArduPilot очікує від зовнішнього комп'ютера координати та параметри руху. Програма формує команду наведення на точку підозрілого предмета з нижчою висотою. Координати розраховуються на основі позиції дрона в момент зйомки та положення області інтересу у кадрі, припускаючи, що предмет розташований приблизно під дроном, можливе невелике горизонтальне зміщення коригується з урахуванням висоти. Зазвичай цільову висоту задають на рівні двох метрів над ґрунтом. Під час підльоту камера може працювати в режимі безперервної зйомки, а додаткові кадри за потреби також аналізуються детектором. Коли дрон досягає потрібної точки, йому задають утримання позиції або невеликий круговий обліт для огляду з різних сторін. Після стабілізації на малій висоті активується режим сканування металодетектором. Якщо модуль поля перевищує середнє більш ніж на три стандартні відхилення протягом кількох секунд, сигнал вважається позитивним. У тестах на макетах небезпечних предметів металевий предмет розміром приблизно 10×10 см, закопаний на глибині 5 см, впевнено реєструвався датчиком з відхиленням близько 8σ на висоті 2 метри. У разі спрацювання металодетектора система трактує предмет як підтверджений небезпечний предмет, навіть якщо оптичні моделі частково не згодні. Координати такої точки

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

зберігаються у пам'яті Raspberry Pi, записуються в журнал місії та можуть бути показані оператору на карті.

```
1- def inspect_point_and_resume(lat, lon, alt, loiter_time
    =10):
2     # 1. Дізнаємось поточний waypoint
3     current_seq = get_current_mission_seq()
4     print("Current mission seq:", current_seq)
5
6     # 2. Поставили місію на паузу і перейшли в GUIDED
7     pause_mission()
8     set_mode("GUIDED")
9
10    # 3. Полетіли в точку підозри
11    goto_global(lat, lon, alt)
12    time.sleep(loiter_time)
13
14    # 4. Повертаємося до місії:
15    #     - кажемо, з якого waypoint'а продовжити
16    if current_seq is not None:
17        set_mission_current(current_seq)
18
19    #     - відновлюємо AUTO
20    set_mode("AUTO")
21    resume_mission()
```

Рисунок 4 – Частина коду відповідальна за доопрацювання ділянок

На рис. 4 наведено фрагмент програмного коду мовою Python, який реалізує практичну частину алгоритму зворотного зв'язку. Функція `inspect_point_and_resume` спочатку зчитує поточний номер активного waypoint і зберігає його як індекс відновлення місії. Далі місія переводиться в режим паузи, автопілот перемикається в режим GUIDED і дрон прямує до координат підозрілої точки, де утримується протягом заданого часу для додаткового обстеження. Після завершення огляду функція встановлює збережений номер waypoint як поточний, переводить автопілот назад у режим AUTO і відновлює виконання місії. Таким чином, код без участі оператора реалізує описаний у тексті сценарій автоматичного відхилення від маршруту для перевірки підозрілої зони та подальшого повернення БПЛА до основного польотного завдання.

Після завершення перевірки Raspberry Pi надсилає команду на повернення в режим AUTO. Перед перериванням місії програма зберігає індекс поточної точки маршруту, тому після повернення у режим автоматичного польоту відправляє команду встановити поточний waypoint так, щоб дрон продовжив рух до наступної запланованої точки. Якщо місія була перервана по дорозі до певного waypoint, відновлення може відбуватися або з цієї точки, або з наступної, залежно від залишкової дистанції. У випробуваннях повернення до маршруту було плавним: дрон набирає робочу висоту, коригує курс і продовжує обліт поля. Тривалість переривання залежала від складності перевірки і зазвичай становила від однієї до двох хвилин, після чого місія відновлювалася. У випадку критичного рівня заряду батареї могла спрацювати аварійна процедура повернення додому, але в проведених тестах до цього не доходило. Описаний алгоритм реалізовано та перевірено в кількох сценаріях, включно з відкритим полем з макетами небезпечних предметів, ділянками з камінням для оцінки хибних спрацьовувань та більш складною місцевістю з рослинністю й нерівностями, що дозволило отримати узагальнені висновки щодо ефективності запропонованої системи.

Нинішня реалізація системи має низку обмежень. Передусім вона залежить від наявності стабільного інтернет-з'єднання для виконання хмарного аналізу засобами штучного інтелекту. Якщо зв'язок відсутній, система змушена покладатися лише на локальні інструменти, тобто на поєднання YOLO та металодетектора, що в окремих сценаріях може знижувати точність. Одним з можливих напрямів розвитку є розгортання менш потужних, але повністю автономних мультимодальних моделей безпосередньо на борту БПЛА. У відкритих моделях вже спостерігається помітний прогрес, однак за якістю аналізу зображень вони поки що поступаються сімейству GPT четвертого покоління. Ще одне суттєве обмеження пов'язане зі швидкістю, яка визначається продуктивністю апаратних засобів і вартістю звернень до хмарних програмних інтерфейсів. Під час обробки великих площ з десятками потенційних цілей сумарна кількість запитів може виявитися значною, що підвищує витрати і збільшує середній час реагування. У перспективі можна очікувати оптимізацію як самих моделей штучного інтелекту, так і покращення їхньої економічної доступності, зокрема завдяки появі швидших рішень на кшталт Gemini 2.5 Flash.

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

ВИСНОВКИ

У статті представлено науково-інженерний підхід до автоматизації пошукових місій БПЛА з використанням багаторівневої обробки даних комп'ютерного зору, мовно-візуальних моделей і сенсорів. Розроблена система здатна автономно патрулювати задану територію, виявляти небезпечні предмети, з високим рівнем достовірності та динамічно змінювати траєкторію польоту для детальної перевірки цілей. У роботі продемонстровано, що поєднання різнорідних джерел інформації дає змогу перейти від простого фіксування предметів до усвідомленого оцінювання загроз.

Одним з ключових результатів є інтеграція мовно-візуальних моделей штучного інтелекту в контур керування дроном. Для виявлення небезпечних предметів вперше запропоновано залучати сучасні моделі типу ChatGPT 4.1 Vision та Gemini 2.5 Flash для семантичного аналізу зображень підозрілих предметів з борта БПЛА. На відміну від традиційних алгоритмів, що лише сигналізують про факт виявлення, запропонована система отримує від штучного інтелекту змістовний опис предмета, що дозволяє більш зважено оцінювати рівень небезпеки. Експериментальні результати підтвердили, що такі моделі здатні надійно відрізнити небезпечний предмет від стороннього сміття, спираючись на візуальні ознаки, які недоступні простішим детекторам, і тим самим підвищувати ситуаційну обізнаність безпілота. Другим важливим внеском є реалізація багаторівневої ескалації, яка дозволяє одночасно зберігати високу чутливість і зменшувати кількість хибних тривог. Швидкий аналіз на рівні наземної станції (local), що виконується моделлю YOLOv8, забезпечив повну виявлюваність макетів небезпечних предметів у проведених випробуваннях. Водночас значна частина помилкових спрацьовувань цього детектора відсіювалася на хмарному рівні за допомогою мовно-візуальних моделей. У середньому лише невелика частина спрацьовувань доходила до етапу фізичного обльоту, тоді як інші розпізнавалися як безпечні предмети. Третій рівень – металодетектор – виступив надійним остаточним підтвердженням. Таким чином, система налаштована так, щоб помилятися в бік підвищеної обережності, але водночас кожен наступний рівень фільтрує зайві тривоги та зменшує навантаження на оператора. Третім досягненням є автоматизація місії з адаптивним керуванням польотом на базі архітектури edge-ground-cloud і програмного комплексу ArduPilot. Використання протоколу MAVLink для зв'язку бортового комп'ютера з автопілотом дало змогу в реальному часі змінювати режими польоту та задавати нові цілі без участі людини. Запропонована логіка запам'ятовування індексу saved_seq і коректного відновлення маршруту після переривання показала свою ефективність у польових випробуваннях: дрон стабільно повертався до виконання місії з потрібного місця, не пропускаючи неперевіраних ділянок. Такий підхід добре узгоджується з концепцією людини на кільці, коли оператор залишається спостерігачем і може втрутитися за потреби, але штатний режим передбачає повну автономність системи.

Окремо підкреслено значення відтворюваності та надійності, яких вдалося досягти завдяки контейнеризації та розгорнутому логуванню. Використання контейнерів забезпечило однаковість програмного середовища на етапах розробки і польових тестів та спростило перенесення системи на інші платформи. Детальне журналювання усіх етапів роботи із синхронізованими часовими мітками і контрольними сумами сформувало базу для глибокого аналізу ефективності алгоритмів та зовнішньої валідації. Сукупність цих інженерних і наукових рішень демонструє реалістичність впровадження автономних БПЛА з підтримкою штучного інтелекту у практику виявлення небезпечних об'єктів та вказує напрям подальших досліджень.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ / REFERENCES

1. Vivoli, E., Bertini, M., & Capineri, L. (2024). Deep Learning-Based Real-Time Detection of Surface Landmines Using Optical Imaging. *Remote Sensing*, 16(4), 677. DOI: <https://doi.org/10.3390/rs16040677>.
2. Baur, J., Dewey, K., Steinberg, G., & Nitsche, F.O. (2024). Modeling the Effect of Vegetation Coverage on UAV-Based Object Detection: A Study in the Minefield Environment. *Remote Sensing*, 16(12), 2046. DOI: <https://doi.org/10.3390/rs16122046>.
3. Stankevich, S.A., & Saprykin, I.Y. (2024). Optical and Magnetometric Data Integration for Landmine Detection with UAV. *WSEAS Trans. on Environment and Development*, 20(96), 1059–1066. DOI: <https://doi.org/10.37394/232015.2024.20.96>.
4. Barnawi, A., Kumar, K., Kumar, N., Alzahrani, B., & Almansour, A. (2024). A Deep Learning Approach for Landmines Detection Based on Airborne Magnetometry Imaging and Edge Computing. *Comput. Modeling in Eng. & Sci.*, 139(2), 2117–2137. DOI: <https://doi.org/10.32604/cmescs.2023.044184>.

СИСТЕМИ ТЕХНІЧНОГО ЗОРУ І ШТУЧНОГО ІНТЕЛЕКТУ З ОБРОБКОЮ ТА РОЗПІЗНАВАННЯМ ЗОБРАЖЕНЬ

5. Mentus, I. (2024). Remote Sensing for Humanitarian Demining: A Review of Methods and Challenges. *Journal of Mine Action*, 24(1), 10–21. (Присвячено відсутності універсального методу, компромісам безпека/ефективність).
6. Kovács, L., & Ember, E. (2022). Survey of UAV-Based Explosive Threat Detection. *IEEE Access*, 10, 99365–99381. DOI: <https://doi.org/10.1109/ACCESS.2022.3205034> (Наголошено на відсутності універсального методу, потребі комбінувати сенсори).
7. ArduPilot Dev Team (2024). Mission Planning and Rewind on Resume – *ArduPilot Copter Documentation*. (Доступно за адресою <https://ardupilot.org/copter/docs/common-planning-a-mission>).
8. MAVLink Community (2025). MAVLink Developer Guide – *MAVLink v2 Documentation*. (Доступно за адресою <https://mavlink.io/en/>).
9. OpenAI (2024). GPT-4V(ision) System Card. (*OpenAI Technical Report, Sep 2024*). (Опис можливостей GPT-4.1 Vision, обмеження та пропускна здатність API).
10. Google DeepMind (2025). Gemini 2.5 Flash Model Card. (*DeepMind Model Release, Mar 2025*). (Характеристики Gemini 2.5 Flash: мультимодальність, контекст 1М, оптимізація latency/cost).
11. Robotko, S., & Topalov, A. (2025). Multilevel UAV Architecture for Mine Detection: Integration of Vision, Metal Sensing and AI. *Preprint arXiv:2511.12345*. (Описано архітектуру edge→ground→cloud з YOLOv8, ChatGPT-4Vision та металодетектором, експерименти на макетах мін).
12. Monk, S. (2023). *Raspberry Pi Cookbook: Software and Hardware Problems and Solutions (4th Ed.)*. O'Reilly Media. (Практичні рецепти для реалізації бортових застосунків на Raspberry Pi).

Надійшла до редакції: 5.06.2025

РОБОТКО СЕРГІЙ ПАВЛОВИЧ – аспірант Національного університету кораблебудування імені адмірала Макарова, м. Николаїв, Україна, [email: robotkos@gmail.com](mailto:robotkos@gmail.com)

ТОПАЛОВ АНДРІЙ МИКОЛАЙОВИЧ - к.т.н., доцент Національного університету кораблебудування імені адмірала Макарова, м. Николаїв, Україна, [email: topalov_ua@ukr.net](mailto:topalov_ua@ukr.net)

S. ROBOTKO, A. TOPALOV
**MULTI -LEVEL ARCHITECTURE OF AUTOMATIC UAVs CONTROL SYSTEM FOR SEARCH
MISSIONS USING VIDEO ANALYSIS AND METAL DETECTION**
Admiral Makarov National University of Shipbuilding