Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Донавчання та оптимізація нейромережевої системи детекції БПЛА для вбудованих обчислювачів засобів перехоплення
(тема)

Виконав:
здобувач четвертого року навчання,
групи ІТШІ-22-2
Анна КНИШ
(власне ім'я, прізвище)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма Штучний інтелект
(повна назва освітньої програми)
Керівник доц. Марія ГОЛОВЯНКО
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Лариса ЧАЛА
(підпис) (власне ім'я, прізвище)

2026 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Штучний інтелект
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Книш Анні Олексіївні
(прізвище, ім'я, по батькові)

1. Тема роботи Донавчання та оптимізація нейромережевої системи детекції БПЛА для вбудованих обчислювачів засобів перехоплення

затверджена наказом університету від 5 червня 20 26 р. № 488Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 14 липня 20 26 р.

3. Вихідні дані до роботи Науково-технічні публікації в галузі комп'ютерного зору та супроводження БПЛА в реальному часі, технічна документація мови Python, бібліотек PyTorch, OpenCV, Ultralytics, моделей SAM 3, DINOv2 та інструментарію NVIDIA TensorRT, специфікації обчислювальних платформ Edge AI (зокрема NVIDIA Jetson, Orange Pi та Raspberry Pi), відеоматеріали бойової роботи БПЛА з відкритих OSINT-джерел, набір графічних даних із зображеннями військових БПЛА літакового типу та негативних зразків для навчання й тестування моделей.

4. Перелік питань, що потрібно опрацювати в роботі

1) Аналіз предметної галузі та апаратного забезпечення систем перехоплення

2) Дослідження методів машинного навчання та оптимізації нейромереж

3) Методологія підготовки даних

4) Експериментальні дослідження, оптимізація та аналіз результатів

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	08.06.2026	виконано
2	Аналіз предметної галузі та апаратних обмежень	12.06.2026	виконано
	Edge-обчислювачів		
3	Дослідження методів машинного навчання,	15.06.2026	виконано
	детекції та трекінгу цілей		
4	Збір, напівавтоматизована розмітка та	19.06.2026	виконано
	дедуплікація набору даних		
5	Експериментальні дослідження, апаратна	25.06.2026	виконано
	оптимізація та квантизація моделей		
6	Написання пояснювальної записки	29.06.2026	виконано
7	Перевірка на академічний плагіат	30.06.2026	виконано
8	Нормоконтроль	06.07.2026	виконано
9	Підготовка презентації та доповіді	08.07.2026	виконано
10	Попередній захист	10.07.2026	виконано
11	Рецензування	11.07.2026	виконано
12	Захист перед ЕК	14.07.2026	

Дата видачі завдання 8 червня 2026 р.

Здобувач _____
(підпис)

Керівник роботи _____ доц. Марія ГОЛОВЯНКО
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 139 с., 5 рис., 13 табл., 9 дод., 26 джерел.

ДЕДУПЛІКАЦІЯ ДАНИХ, ДЕТЕКЦІЯ БПЛА, КВАНТИЗАЦІЯ, КОМП’ЮТЕРНИЙ ЗІР, ТЕМПОРАЛЬНЕ СУПРОВОДЖЕННЯ, ШТУЧНИЙ ІНТЕЛЕКТ, BYTETRACK, OSINT-ДЖЕРЕЛА, TENSORRT, YOLO26N.

Об’єкт дослідження – процес автоматизованої детекції та відстеження безпілотних літальних апаратів у реальному часі.

Предмет дослідження – методи донавчання та апаратно-програмної оптимізації нейромережевих моделей комп'ютерного зору для роботи на ресурсно-обмежених обчислювачах.

Мета роботи – розробити та оптимізувати конвеєра комп'ютерного зору для детекції та темпорального супроводження військових БПЛА на базі вбудованих обчислювальних платформ.

Методи дослідження – комп'ютерний зір, алгоритмічна оптимізації, методи обчислювального експерименту та статистичного аналізу.

У роботі розроблено програмний конвеєр комп'ютерного зору для напівавтономного перехоплення БПЛА. Для побудови навчального набору реалізовано збір та розмітку кадрів моделями Gemini 2.5 Pro та SAM 3 із дедуплікацією за IoU та ембедингами DINOv2 (алгоритм k-Center Greedy), що дало змогу сформувати датасет обсягом 13 200 кадрів. Навчено модель YOLO26n, компіляція якої у TensorRT FP16 забезпечила швидкодію 95,7 FPS (затримка 10,45 мс) при деградації mAP50 лише на 0,01%. Аналіз трекерів підтвердив перевагу ByteTrack над OC-SORT в умовах шумних детекцій (HOTA – 69,26%, IDF1 – 84,14%) при затримці 0,568 мс, а наскрізна затримка обробки кадру конвеєром склала 13,14 мс (76 FPS), що доводить працездатність системи.

ABSTRACT

Bachelor’s thesis contains: 139 pp., 5 fig., 13 tabl., 9 ann., 26 references.

ARTIFICIAL INTELLIGENCE, BYTETRACK, COMPUTER VISION, DATA DEDUPLICATION, OSINT SOURCES, QUANTIZATION, TEMPORAL TRACKING, TENSORRT, UAV DETECTION, YOLO26N.

Object of research – the process of real-time automated detection and tracking of unmanned aerial vehicles.

Subject of research – methods for fine-tuning and hardware-software optimization of computer vision neural network models to run on resource-constrained computing platforms.

Goal of research – to develop and optimize a computer vision pipeline for the detection and temporal tracking of military UAVs on embedded computing platforms.

Research methods – computer vision, algorithmic optimization, methods of computational experiment and statistical analysis.

This study presents a real-time computer vision pipeline for semi-autonomous UAV interception. Video frames annotated using Gemini 2.5 Pro and SAM 3 models were deduplicated based on IoU and DINOv2 semantic embeddings (via the k-Center Greedy algorithm), yielding a verified dataset of 13,200 frames. The YOLO26n model compiled into TensorRT FP16 achieved 95.7 FPS (10.45 ms latency) with a minor mAP50 degradation of only 0.01%. Tracker comparison confirmed the superiority of ByteTrack over OC-SORT under noisy conditions (69.26% HOTA, 84.14% IDF1) at 0.568 ms update latency, while the 13.14 ms (76 FPS) end-to-end pipeline latency demonstrates the system's practical viability.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів 10
Вступ 12
1 Аналіз предметної галузі та апаратного забезпечення систем перехоплення 14
1.1 Формування тактико-технічних вимог до апаратної бази перехоплювачів 14
1.2 Аналіз характеристик та ресурсних обмежень edge-обчислювачів як цільової апаратної бази 16
1.3 Визначення референсних апаратних обмежень для оцінювання швидкодії системи 17
2 Дослідження методів машинного навчання та оптимізації нейромереж . 19
2.1 Аналіз вимог до підсистем детекції та трекінгу і відбір релевантних нейромережевих архітектур 19
2.1.1 Аналіз та відбір нейромережевих архітектур детекції 20
2.1.2 Аналіз та відбір алгоритмів трекінгу 22
2.2 Теоретичне обґрунтування методології збору та підготовки даних для донавчання 23
2.2.1 Обґрунтування необхідності створення спеціалізованої вибірки 23
2.2.2 Аналіз джерел та формування вимог до навчального набору ... 26
2.2.3 Методологія напівавтоматичного збору даних 29
2.2.4 Методологія розмітки та стратегія монокласової детекції 31
2.2.5 Теоретичне обґрунтування обмежень та відмови від стандартних стратегій аугментації 32
2.2.6 Запобігання витоку даних та поділ вибірки 33
2.2.7 Метрики оцінки якості датасету 34
2.2.8 Обґрунтування цільового обсягу вибірки 35

2.3 Дослідження методів оптимізації моделей глибокого навчання для Edge-пристроїв 36
2.3.1 Прунінг 37
2.3.2 Квантизація 38
2.3.3 Компіляція та апаратне прискорення 39
3 Методологія підготовки даних 40
3.1 Методологія збору та джерела первинних OSINT-даних 40
3.2 Первинна фільтрація та критерії вибракування вихідних файлів 42
3.3 Ручна сегментація та виділення поодиноких епізодів збиття 44
3.4 Автоматизована класифікація метаданих за допомогою VLM 46
3.5 Повторна фільтрація крос-джерельних дублікатів відеоепізодів 49
3.6 Аналіз характеристик та структурного розподілу набору даних 51
3.7 Методологія стратифікованого розподілу вибірки та запобігання витоку даних 56
3.8 Напівавтоматизоване анотування, верифікація та структурування даних 58
3.8.1 Оцінка складності та обґрунтування вибору гібридного підходу 59
3.8.2 Генерація псевдоанотацій за допомогою моделі SAM 3 60
3.8.3 Коригування анотацій у середовищі CVAT 61
3.8.4 Екстракція кадрів та конвертація анотацій у формат YOLO 63
3.8.5 Методологія інтеграції та синтетичного моделювання негативних зразків 65
3.9 Триетапний алгоритм дедуплікації даних на рівні розмічених кадрів 66
3.9.1 Обмеження класичних методів дедуплікації 67
3.9.2 Просторова дедуплікація за метрикою IoU 69
3.9.3 Семантична дедуплікація на основі зважених ембедингів 70
3.9.4 Глобальна семантична дедуплікація на основі зважених ембедингів 73

3.10 Статистичний аналіз та верифікація дедуплікованого набору даних 75
3.10.1 Кількісний аналіз та розподіл даних за вибірками..... 75
3.10.2 Семантичний розподіл метаданих дедуплікованих кадрів 76
3.10.3 Геометричний аналіз обмежувальних рамок цілей..... 78
3.10.4 Просторова верифікація центрів цілей 79
3.10.5 Математичний аналіз локального контрасту цілі відносно тла 80
4 Експериментальні дослідження, оптимізація та аналіз результатів 83
4.1 Конфігурація експериментальних досліджень та методологія пошуку навчання за спрощеними шляхами..... 83
4.2 Аналіз карт активації EigenCAM та дослідження помилкових кореляцій 85
4.3 Конфігурація процесу оптимізації та аналіз динаміки навчання моделі 88
4.4 Апаратно-програмна оптимізація та низькорівнева квантизація моделі детекції 91
4.4.1 Порівняльний аналіз PyTorch FP32 та TensorRT FP16 92
4.4.2 Аналіз накладних витрат препроцесингу та рівнів швидкодії.. 94
4.4.3 Системні обмеження та обґрунтування відмови від INT8-квантизації 95
4.4.4 Методологічні обмеження симуляції та екстраполяції на..... 96
4.5 Аналіз хибнопозитивних та хибнонегативних спрацювань у розрізі семантичних доменів 97
4.6 Інтеграція та порівняльний аналіз алгоритмів темпорального трекінгу 105
4.7 Теоретична оцінка наскрізної швидкодії програмного конвеєра 110
Висновки 113
Перелік джерел посилання 115

Додаток А Лістинг програми автоматичного маркування метаданих VLM 118
Додаток Б Програмна реалізація алгоритму напівавтоматичної розмітки даних 121
Додаток В Програмна реалізація алгоритму дедуплікації за метрикою IoU 123
Додаток Г Програмна реалізація алгоритму семантичної дедуплікації даних 125
Додаток Д Програмна реалізація конвеєра навчання та тестування моделі детекції 128
Додаток Е Лістинг програми компіляції моделі YOLO26n та вимірювання швидкодії інференсу 130
Додаток Ж Лістинг програми розрахунку карт активації EigenCAM 132
Додаток И Лістинг програми порівняльного тестування та вимірювання затримок трекерів ByteTrack та OC-SORT 135
Додаток К Відомість кваліфікаційної роботи 139

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БПЛА – безпілотний літальний апарат;

НСК – наземна станція керування;

РЕБ – радіоелектронна боротьба;

AI – Artificial Intelligence – штучний інтелект;

ASIC – Application-Specific Integrated Circuit – спеціалізована мікросхема для конкретного застосування;

CPU – Central Processing Unit – центральний процесор;

CVAT – Computer Vision Annotation Tool – інструмент розмітки даних комп'ютерного зору;

DFL – Distribution Focal Loss – функція втрат розподілу локалізації;

DINOv2 – нейромережа для отримання універсальних самокерованих семантичних ембедингів зображень;

FPS – Frames Per Second – кількість кадрів за секунду;

GPU – Graphics Processing Unit – графічний процесор;

HOTA – Higher Order Tracking Accuracy – метрика оцінки якості темпорального супроводження;

IDF1 – ID F1-score – метрика стабільності утримання унікального ідентифікатора траєкторії;

IoU – Intersection over Union – метрика перетину обмежувальних рамок);

IR – Infrared (інфрачервоний тепловізійний спектр;

mAP – Mean Average Precision – середня точність роботи моделі детекції;

MOT – Multiple Object Tracking – формат представлення даних відстеження кількох об'єктів;

MOTA – Multiple Object Tracking Accuracy – метрика точності супроводження множини об’єктів;

NIR – Near-Infrared – ближній інфрачервоний спектр;

NMS – Non-Maximum Suppression – алгоритм пригнічення не-максимумів;

NPU – Neural Processing Unit – нейропроцесор, прискорювач матричних обчислень;

OSD – On-Screen Display – інтерфейсна графіка польотної телеметрії, що накладається на відеопотік;

OSINT – Open Source Intelligence – розвідка та збір даних на основі відкритих джерел;

RGB – Red, Green, Blue – денний кольоровий спектр камери;

SAM 3 – Meta Segment Anything Model 3 – мультимодальна нейромережа сегментації об'єктів;

SBC – Single Board Computer – одноплатний комп'ютер;

SWaP – Size, Weight, and Power – масогабаритні та енергетичні обмеження вбудованих систем;

VLM – Vision-Language Models – мультимодальні візуально-мовні моделі;

YOLO – You Only Look Once – сімейство архітектур нейромереж для одноетапної детекції об'єктів у реальному часі.

ВСТУП

Сучасний етап розвитку безпілотних технологій характеризується різким збільшенням ролі БПЛА різного класу в бойових діях. Особливо критичною є загроза від розвідувальних безпілотників літакового типу та баражуючих боєприпасів. Традиційні засоби протидії часто є економічно недоцільними або неефективними проти малорозмірних цілей, що маневрують на низьких висотах. У зв'язку з цим високу актуальність має розробка автономних дронів-перехоплювачів, здатних виявляти та наводитися на ворожі БПЛА в автоматичному режимі.

Інтеграція систем комп'ютерного зору на борт легких перехоплювачів обмежена жорстким апаратним бюджетом та необхідністю забезпечення частоти обробки відеопотоку не менше 30 кадрів за секунду. Наявні у відкритому доступі набори даних розраховані на статичну наземну перспективу та цивільні квадрокоптери, що робить їх непридатними для умов повітряного бою. Окрім того, використання відеоматеріалів із відкритих джерел супроводжується наявністю водяних знаків та елементів телеметрії, що за умов притаманної нейромережам упередженості до простоти провокує виникнення ефекту «Розумного Ганса».

Це зумовлює наукову та практичну потребу в розробці спеціалізованої методології підготовки даних, навчанні легкої моделі детекції та її низькорівневій апаратній оптимізації для надійної роботи в режимі реального часу.

Мета роботи – розробка та дослідження високопродуктивного програмного конвеєра комп'ютерного зору для детекції та темпорального супроводження військових БПЛА літакового типу в реальному часі на базі вбудованих обчислювальних платформ.

Для досягнення поставленої мети необхідно вирішити такі завдання:

– сформулювати тактико-технічні вимоги та апаратні обмеження бортових систем перехоплення БПЛА;

– провести порівняльний аналіз сучасних архітектур детекції та алгоритмів трекінгу, обґрунтувати вибір оптимальних базових моделей;

– розробити методологію збору, первинної фільтрації та структурування завадостійкого OSINT-датасету;

– проєктувати та реалізувати автоматизований конвеєр напівавтоматичної розмітки кадрів за допомогою моделі Segment Anything (SAM 3) та хмарних середовищ анотування;

– розробити математичні методи усунення надмірності відеоданих на основі семантичного аналізу та просторової дедуплікації;

– навчити модель YOLO26n, виконати діагностику її просторової уваги за допомогою карт активації EigenCAM та провести низькорівневу оптимізацію обчислювального графа в TensorRT;

– експериментально оцінити затримки та швидкодію всього програмного конвеєра детекції та трекінгу.

Об’єкт дослідження – процес автоматичної детекції та безперервного темпорального супроводження малорозмірних повітряних об'єктів у режимі реального часу на ресурсообмежених бортових обчислювачах.

Предмет дослідження – математичні моделі, алгоритми, методи оптимізації обчислювальних графів нейромереж та методологія формування, очищення і регуляризації навчальних вибірок комп'ютерного зору.

Оптимізовано обчислювально легкий програмний конвеєр детекції та супроводження БПЛА на основі моделей YOLO26n та ByteTrack. Наскрізна затримка обробки одного кадру конвеєром становить 13,14 мс (що еквівалентно 76 FPS), що гарантує надійне функціонування системи у режимі реального часу на вбудованому обладнанні та залишає понад 20 мс вільного часу на кадр для розрахунку кінематики польотним контролером. Створений верифікований датасет обсягом 13 200 кадрів та інструменти автоматизації розмітки розміщені у відкритому доступі для подальших досліджень.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМ ПЕРЕХОПЛЕННЯ

Для розробки системи напівавтономного перехоплення БПЛА необхідним є врахування умов експлуатації та апаратних обмежень платформи-носія. У цьому розділі проаналізовано специфіку бойового застосування дронів-перехоплювачів, сформульовано тактико-технічні вимоги до підсистем комп'ютерного зору, досліджено співвідношення вартості та продуктивності сучасних Edge-обчислювачів та проведено порівняльний аналіз їхніх характеристик для розгортання нейромережевих моделей.

1.1 Формування тактико-технічних вимог до апаратної бази перехоплювачів

Аналіз характеристик БПЛА свідчить про екстремальну динаміку повітряного бою. Швидкість гвинтових цілей типу Орлан-10 або Shahed-136 становить 65–185 км/год, а реактивні модифікації Shahed-238 розвивають до 600 км/год, що перевищує 166 м/с [1]. Для гарантованого перехоплення вітчизняні платформи X2 Jet та Багнет-АА спроєктовано зі швидкостями до 426 км/год або 118 м/с [2], [3]. При зустрічному зіткненні сумарна швидкість зближення $V_{\text{зближення}}$ сягає майже 170 м/с для гвинтових і до 280 м/с для реактивних цілей.

У таких умовах алгоритмічна затримка обробки відеопотоку стає найкритичнішим параметром. Просторове зміщення цілі відносно перехоплювача за один кадр визначається співвідношенням:

$$\Delta S = \frac{V_{\text{зближення}}}{FPS}, \tag{1.1}$$

де ΔS – просторове зміщення цілі відносно перехоплювача за один кадр, м;

$V_{\text{зближення}}$ – сумарна швидкість зближення цілі та перехоплювача, м/с;

FPS – частота оновлення кадрів відеопотоку, кадр/с.

У найгіршому сценарії при швидкості зближення 280 м/с робота системи на частоті 10 FPS означає затримку у 100 мс, що призводить до стрибкоподібного зміщення об'єкта на 28 метрів за кадр. На типових дистанціях захоплення цілі в 100–300 м кутовий розмір БПЛА у кадрі не перевищує кількох десятків пікселів, тому таке зміщення повністю виводить ціль за межі її попередньої обмежувальної рамки та обнуляє показник Intersection over Union (IoU) між суміжними кадрами. Це гарантовано руйнує роботу алгоритмів трекінгу без використання ресурсомістких ReID-модулів і призводить до втрати цілі у вузькому полі зору камери.

Відповідно, мінімально допустима швидкодія системи має складати 30 FPS, що зберігає перетин обмежувальних рамок для стабільної роботи трекера. Для ефективного ураження маневрових цілей цільовим показником є 60 FPS, що додатково забезпечує необхідну дискретизацію керуючих сигналів для стабільної роботи PID-регуляторів без паразитних осциляцій. Апаратні рішення з продуктивністю нижче 30 FPS є технічно непридатними.

Окрім жорстких вимог до швидкодії, бортова електроніка обмежена суворими масогабаритними та енергетичними параметрами, відомими як SWaP-обмеження. Оскільки встановлення масивних систем охолодження критично погіршує аеродинаміку перехоплювачів, головним завданням дослідження є глибока програмно-апаратна оптимізація нейромережі для досягнення її апаратної агностичності.

Для розробки та донавчання нейромережевої моделі визначальними є програмно-апаратні обмеження, які диктуються реальними умовами бойового застосування.

1.2 Аналіз характеристик та ресурсних обмежень edge-обчислювачів як цільової апаратної бази

Для вибору оптимальної апаратної платформи перехоплювача необхідно проаналізувати наявні на ринку рішення для периферійних обчислень. Згідно з дослідженнями розвитку апаратних ШІ-архітектур, їх можна класифікувати за трьома основними парадигмами [4, с. 3]:

– базові обчислювачі без спеціалізованих прискорювачів. Виконання матричних обчислень у таких системах покладається виключно на ядра архітектури ARM, що характерно для базових конфігурацій Raspberry Pi. Через відсутність апаратного розпаралелювання цей клас пристроїв демонструє пропускну здатність на рівні одиничних кадрів [4, с. 4]. Такі показники є технічно непридатними для задач реального часу, тому ці обчислювачі виключаються з розгляду;

– модульні гетерогенні системи. Цей клас рішень поєднує одноплатний комп'ютер, або SBC, із зовнішнім спеціалізованим нейроприскорювачем типу VPU або ASIC, підключеним через інтерфейси USB або шину PCIe. Типовим прикладом є зв'язка Raspberry Pi з модулями Intel Neural Compute Stick 2 або Google Coral Edge TPU. Головним викликом для таких систем є обмежена пропускна здатність шини передачі даних між оперативною пам'яттю плати та власною пам'яттю прискорювача. Використання застарілих інтерфейсів USB створює критичні затримки, проте інтеграція через високошвидкісні шини PCIe версій 2.0 або 3.0 дозволяє ефективно розвантажити центральний процесор для виконання завдань керування польотом;

– інтегровані системи на кристалі з вбудованим NPU або GPU. Це найбільш ефективна архітектура, у якій центральний процесор та апаратний прискорювач розміщені на одному кристалі та використовують спільну пам'ять. Таке рішення мінімізує затримки передачі даних.

Останній клас інтегрованих систем поділяється на два підтипи:

– GPU-SoC представлені лінійкою NVIDIA Jetson. Вони використовують масивну паралельну архітектуру з ядрами CUDA або Tensor. Ці платформи забезпечують високу продуктивність, але традиційно мають значний тепловий пакет від 10–15 Вт та високу вартість [5];

– NPU-SoC, прикладом яких є лінійка Rockchip RK3588S. Вони поєднують енергоефективні ARM-ядра із вбудованими нейропроцесорами типу ASIC або NPU, шо дозволяє досягти цільової продуктивності близько 6 TOPS при рекордній ефективності обчислень на ват енергії, що є критично важливим для автономних систем [6, с. 11].

1.3 Визначення референсних апаратних обмежень для оцінювання швидкодії системи

У результаті аналізу ринку визначено три референсні енергоефективні архітектури ARM, які задають граничні техніко-економічні рамки для проєктування системи. Їхні техніко-економічні характеристики, що виступатимуть обмежувальними орієнтирами для аналізу продуктивності розроблюваного конвеєра, подано в таблиці 1.1.

Таблиця 1.1 – Техніко-економічні параметри референсних Edge-платформ

Модель платформи	Продуктивність, TOPS	Обсяг RAM, ГБ	Енергоспоживання, Вт	Вартість, $
NVIDIA Jetson Orin Nano 4GB	10	4	7 – 10	260
Orange Pi 5	6	8	5 – 10	159
Raspberry Pi 5 (8GB) + Hailo-8L	13	8	12 – 14	245

Примітка 1. Вартість апаратних комплектів (включно з активним/пасивним охолодженням) вказано за базовими цінами дистриб'юторів.

Примітка 2. Продуктивність Jetson вказана у консервативному форматі Dense, тобто обчислення на щільних матрицях, для коректного порівняння з класичними NPU;

Аналізуючи систематизовані дані, наведені у таблиці 1.1, можна зробити обґрунтований висновок, що обрані цільові бюджетні обчислювальні платформи здатні забезпечити апаратну продуктивність у межах 6–13 TOPS при доступному обсязі оперативної пам’яті від 4 до 8 ГБ. Саме ці ключові технічні характеристики є визначальними критеріями для вибору фінальної архітектури нейромережі. Проектована модель має бути обчислювально легкою, повністю вкладатися в апаратні ліміти платформ і забезпечувати високу швидкість реального інференсу (не менше 30 FPS), необхідну для стабільного функціонування системи автоматичного супроводження.

2 ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ТА ОПТИМІЗАЦІЇ НЕЙРОМЕРЕЖ

2.1 Аналіз вимог до підсистем детекції та трекінгу і відбір релевантних нейромережевих архітектур

Програмний комплекс бортового комп'ютерного зору працює як двокомпонентний конвеєр реального часу. Підсистема просторової детекції виконує покадрове розпізнавання цілі та формує обмежувальні рамки з координатами. Через схильність нейромереж до візуального тремтіння та пропусків детекції на складному фоні результати передаються до підсистеми темпорального трекінгу. Трекер згладжує траєкторію, усуває шуми та асоціює ціль між кадрами для формування чистого вектора наведення.

З огляду на апаратні обмеження платформ, до компонентів конвеєра висуваються такі жорсткі вимоги:

– загальний часовий бюджет одного кадру при 30 FPS становить 33.3 мс. Оскільки детектор є найбільш обчислювально навантаженим компонентом конвеєра, переважна частина цього бюджету має відводитися саме на інференс, залишаючи резерв для підсистеми супроводження та передачі телеметрії польотному контролеру;

– архітектура має гарантовано підтримувати цілочисельну квантизацію в INT8. Згідно з дослідженнями робастності моделей сімейства YOLO до INT8-компіляції у середовищі TensorRT [7, с. 5], перехід до цілочисельного формату збереження зумовлює деградацію метрики детекції у межах 3-7% абсолютного mAP50-95, причому менші архітектури зазнають більшого впливу. Такий рівень похибки є обґрунтованим компромісом, оскільки дозволяє досягти прискорення інференсу в 1.5–3.3 раза порівняно з FP32-оригіналом [7, с. 5], що є критичною умовою для

виконання вимоги у 30 FPS на ресурсообмежених Edge-системах без руйнування контуру автонаведення.

Оскільки апаратний прискорювач повністю задіяний детектором, трекінг виконується на центральному процесорі, тому алгоритм супроводження повинен мати мінімальну обчислювальну складність – без нейромережевих компонентів та операцій вилучення ознак, щоб зберегти резерв CPU для розрахунку кінематики наведення та передачі команд польотному контролеру.

2.1.1 Аналіз та відбір нейромережевих архітектур детекції

Оцінка сучасних парадигм комп'ютерного зору крізь призму жорстких апаратних обмежень дозволила відсіяти архітектури, що є фундаментально непридатними для використання у вбудованих системах. Зокрема, двоетапні детектори виключено через низьку швидкодію та структурну слабкість модуля RPN під час розпізнавання малорозмірних цілей. Візуальні трансформери також визнані недоцільними через квадратичну обчислювальну складність механізму уваги, що робить їх значно важчими за CNN-аналоги.

Відповідно, стандартом для бортових систем залишаються одноетапні згорткові детектори сімейства YOLO. У процесі відбору архітектур YOLOv9 була відхилена через надмірну обчислювальну складність, YOLOv10 – через деградацію метрик при INT8-квантизації, а YOLOv12 – через відсутність ефективного апаратного прискорення її механізмів уваги на Edge-платформах.

Враховуючи обмеження обчислювального бюджету референсних платформ, у межах цього дослідження за замовчуванням розглядаються виключно ультралегкі конфігурації моделей класу Nano. Це дозволяє мінімізувати навантаження на апаратну частину та забезпечити максимальну швидкість обробки кадрів. З огляду на це, для порівняльного

аналізу було обрано дві найсучасніші архітектури цього класу: YOLO11n та YOLO26n.

Модель YOLO11 розглядається як еталонний варіант завдяки високій передбачуваності результатів квантизації, швидкодії, точності та малій кількості параметрів. Проте основним об’єктом дослідження в цій роботі визначено архітектуру YOLO26, спроєктовану спеціально для ресурсно-обмежених вбудованих систем.

У цій моделі усунуто модуль DFL, оскільки операції Softmax не мають ефективного INT8-еквівалента на нейропроцесорах, що зазвичай призводить до колапсу точності. Крім того, YOLO26 реалізовано без використання модуля NMS, що мінімізує загальну латентність. Завдяки впровадженню технологій STAL та ProgLoss, нано версія цієї моделі демонструє вищу точність на датасеті COCO (40,9% проти 39,5% у YOLO11n) при меншій обчислювальній складності (5,4 проти 6,5 GFLOPs) [8], [21].

Теоретичні характеристики обраних конфігурацій Nano-класу наведено в таблиці 2.1.

Таблиця 2.1 – Теоретичні характеристики обраних конфігурацій Nano-класу

Архітектура	Версія	Кількість параметрів, млн	Складність, GFLOPs
YOLO11	Nano (n)	~2,6	~6,5
YOLO26	Nano (n)	~2,5	~5,4

Порівняльний аналіз теоретичних характеристик довів перевагу архітектури YOLO26n. Завдяки меншій обчислювальній складності та меншій кількості параметрів при вищій точності, саме модель YOLO26n була обрана як цільовий детектор для практичного навчання та оптимізації в межах цієї роботи.

Більші конфігурації моделей (класів Small та Medium) були виключені з розгляду, оскільки ризик перевантаження пам'яті та значне споживання ресурсів ускладнюють виконання жорсткої вимоги щодо забезпечення стабільної частоти обробки відеопотоку у 30 FPS за технічних обмежень референсних плат.

2.1.2 Аналіз та відбір алгоритмів трекінгу

Вибір алгоритму супроводження обмежений вимогою мінімального навантаження на центральний процесор, оскільки апаратний прискорювач повністю задіяний детектором, що унеможливлює використання нейромережевих компонентів для вилучення ознак.

Аналіз наявних класів архітектур дозволив відсіяти непридатні рішення. Кореляційні фільтри несумісні з парадигмою Tracking-by-Detection через відсутність механізму відновлення втраченого треку. Трансформерні трекери вимагають значних обчислювальних ресурсів для механізму уваги, що несумісно з вимогою мінімального навантаження на CPU. Алгоритми з Re-ID модулями вимагають окремої нейромережі для вилучення ознак, що на бортових обчислювачах із обмеженим CPU-ресурсом створює неприйнятну затримку навіть при відстеженні одиночної цілі.

Єдиною прийнятною парадигмою залишаються легкі алгоритми на базі IoU-асоціації та фільтра Калмана. Однак базовий SORT ігнорує детекції з низьким рівнем впевненості, що викликає фрагментацію траєкторії при супроводженні малорозмірних цілей на складному фоні [9, с. 1].

Для подолання цих обмежень обрано два вдосконалених алгоритми. Основним кандидатом є ByteTrack, який використовує метод двопрохідної асоціації BYTE. Цей підхід дозволяє аналізувати детекції з низьким показником впевненості на другому проході, що забезпечує відновлення об’єктів, які були б втрачені стандартними методами [10, с. 3]. Як альтернатива розглядається OC-SORT, що коригує стан фільтра Калмана з

урахуванням реальних спостережень у момент відновлення треку. Це забезпечує вищу теоретичну стійкість системи під час різких аеродинамічних маневрів перехоплювача [11, с. 1].

Таким чином, програмний комплекс візуального наведення спроєктовано як двокомпонентний конвеєр із загальним часовим бюджетом 33 мс на кадр. Для підсистеми детекції затверджено модель YOLO26n, що забезпечують баланс між точністю та ефективністю квантизації. Підсистема супроводження на базі ByteTrack або OC-SORT гарантує стабільну частоту відеопотоку 30 FPS завдяки раціональному розподілу навантаження між апаратним прискорювачем та центральним процесором.

2.2 Теоретичне обґрунтування методології збору та підготовки даних для донавчання

Надійність нейромережевої детекції БПЛА залежить від репрезентативності навчальних даних. Специфіка бойових умов вимагає подолання доменного зсуву, оскільки невідповідність між лабораторними умовами навчання та реальним операційним середовищем призводить до деградації точності алгоритмів, тому розробка наукової методології формування спеціалізованої вибірки є критичною для забезпечення точності системи.

2.2.1 Обґрунтування необхідності створення спеціалізованої вибірки

Використання нейромережевих архітектур, попередньо навчених на універсальних наборах даних типу MS COCO, є неефективним для задач термінального наведення. Основною перешкодою є доменний зсув, що виникає через глибоку структурну невідповідність між навчальними

вибірками та реальним бойовим середовищем. Ця проблема проявляється у двох визначальних аспектах.

Першим чинником є екстремальна динаміка масштабів. У межах одного відеопотоку на етапі виявлення ціль займає лише кілька пікселів, проте на термінальній стадії вона стрімко розширюється на весь кадр. Базові моделі не адаптовані до такої різкої зміни масштабів. Другий аспект полягає у суттєвій семантичній похибці класів. Візуальні сигнатури бойових засобів під час інтенсивного маневрування на тлі землі чи хмар кардинально відрізняються від сигнатур цивільних квадрокоптерів, що переважають у відкритих вибірках.

Для підтвердження необхідності формування власного набору даних проаналізовано п'ять найбільш релевантних публічних датасетів [12–16]. Результати порівняння, наведені у таблиці 2.2, свідчать, що всі наявні рішення є непридатними для досліджуваної задачі. Переважна більшість із них використовує або хибну перспективу спостереження, або нерелевантні класи об’єктів.

Таблиця 2.2 – Порівняльний аналіз існуючих публічних датасетів та вимог цільової задачі.

Датасет	Перспектива	Типи БПЛА	Бойові цілі	Повітря-повітря
VisDrone2019	Дрон – земля	–	✗	✗
Anti-UAV	Земля – дрон	DJI, комерційні	✗	✗
DroneVSBird	Земля – дрон	Комерційні	✗	✗
VisioDECT	Земля – дрон	DJI	✗	✗
DroneSOD-30K	Земля – дрон	Комерційні	✗	✗

Примітка. Знак ✗ означає відсутність характеристики у наборі даних.

Рисунок 2.1 унаочнює ці структурні невідповідності. VisDrone2019 фокусується на ракурсі «зверху-вниз» для детекції наземних об'єктів. Anti-UAV, DroneVSBird, VisioDECT та DroneSOD-30K використовують статичну наземну перспективу та обмежуються комерційними мультикоптерами. Цільовий домен характеризується динамічною перспективою «повітря-повітря» та наявністю військових БПЛА літакового типу.

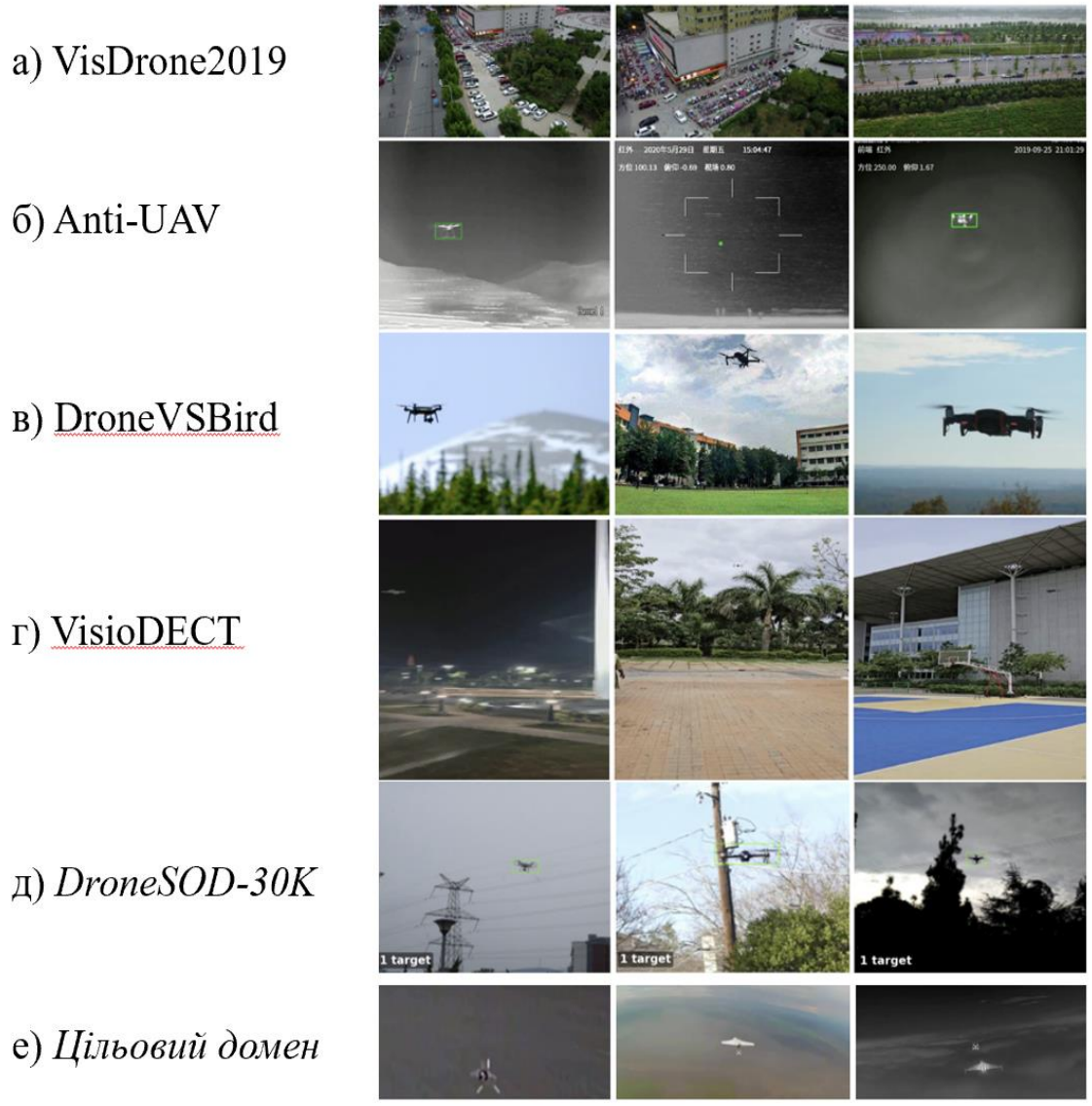


Рисунок 2.1 – Порівняльний аналіз візуальної перспективи та класів цілей у публічних датасетах і цільовому домені

З огляду на це, формування спеціалізованого OSINT-датасету є об'єктивною необхідністю. Тільки такий підхід дозволяє подолати доменний зсув та забезпечити надійну роботу нейромережі в умовах реального повітряного перехоплення.

2.2.2 Аналіз джерел та формування вимог до навчального набору

Розроблюваний комплекс функціонує в парадигмі термінального донаведення, де оператор здійснює первинний пошук цілі, а алгоритм комп'ютерного зору активується лише на фінальній стадії для утримання об'єкта до моменту перехоплення. З огляду на це, до навчального датасету висунуто 6 ключових вимог:

– вибірка має гарантувати репрезентативність усіх типів підстильної поверхні, включаючи небо, хмари, лісосмуги та забудову, що виступають фоном під час атаки з різних кутів;

– набір даних має базуватися виключно на зображеннях реальних військових платформ літакового типу та баражуючих боєприпасів. FPV-камікадзе виключені з вибірки через їхню специфічну кінематику, що потребує окремого дослідження;

– вибірка повинна охоплювати весь спектр розмірів цілі: від моменту захоплення, коли об’єкт займає 1–2% площі кадру, до термінальної стадії, де він перекриває понад 50% зображення;

– кожному екземпляру цілі у відеопослідовностях має бути присвоєно унікальний Track ID. Це необхідно для розрахунку метрик якості супроводження HOTA, MOTA та IDF1;

– до 10% вибірки мають складати кадри без цілі [17]. Обов’язковим є включення кадрів із прицільними марками та елементами телеметрії без БПЛА. Це розриває хибну просторову кореляцію та змушує модель спиратися лише на морфологічні ознаки безпілотника;

– усі типи платформ анотуються єдиним класом Target. Це усуває проблему міжкласового дисбалансу та значно підвищує показник повноти розпізнавання.

Використання відеоматеріалів із відкритих джерел вносить у навчальну вибірку систематичні викривлення, зокрема водяні знаки, елементи інтерфейсу OSD та статичні логотипи. Через притаманну нейромережам упередженість до простоти процес навчання часто спрямовується найлегшим обчислювальним шляхом [18, с. 1–2]. Це створює ефект «Розумного Ганса», коли модель формує хибну статистичну кореляцію між графічним артефактом та об'єктом [19, с. 2–3]. Замість реальних морфологічних ознак безпілотника мережа запам'ятовує контури накладеного тексту чи ліній водяних знаків (рисунок 2.2). В реальних бойових умовах із чистим відеопотоком це призводить до повної втрати здатності детектувати цілі.



Рисунок 2.2 – Приклади критичного перекриття цілі водяними знаками та елементами OSD у відеоматеріалах з відкритих джерел

Для розв'язання цієї проблеми було проаналізовано ряд потенційних методів, частину з яких відхилено з технічних міркувань. Мозаїчна аугментація відхиляється з двох причин: по-перше, вона руйнує просторово-часову зв'язність відеопотоку, унеможливлюючи коректне навчання алгоритмів трекінгу; по-друге, вона не розриває хибну кореляцію між

прицілом (елементом OSD) та ціллю, а лише масштабує і переносить цю стабільну пару в окремі квадранти кадру. Просторовий інпейнтинг та Random Erasing всередині обмежувальних рамок виключено через мікророзмірність цілей на початку відстеження – БПЛА займає лише 1–2% площі кадру. На об'єктах такого масштабу піксельне розмиття ліній прицілу або внесення шуму фізично деформує тонкі градієнти силуету та геометричні контури БПЛА, критично знижуючи повноту детекції. Крім того, локальна інтерполяція утворює мікротекстурні аномалії, які згорткові шари YOLO здатні засвоїти як нові частотні мітки у спектрі Фур'є [20, с. 1 2].

Для усунення зазначених просторових викривлень було теоретично проаналізовано два підходи до регуляризації процесу навчання:

– просторово-обмежене маскування даних (Spatially Constrained Cutout). Метод полягає у випадковому накладанні чорних квадратів виключно на крайові зони кадру, де зазвичай розташовуються статичні логотипи каналів та OSD-текст. Такий підхід динамічно руйнує візуальну цілісність графічних артефактів на кожній епосі навчання, створюючи для згорткової мережі ефект інформаційної порожнечі та примусово фокусуючи увагу глибоких шарів на пошуку справжніх морфологічних ознак БПЛА у центральній частині кадру [21, с. 3]. Попри теоретичну привабливість, впровадження цього алгоритму вимагає глибокої низькорівневої модифікації конвеєра завантаження даних у фреймворку Ultralytics та несе ризики випадкового затінення мікророзмірних цілей у крайових зонах кадру на початкових етапах виявлення;

– інтеграція негативних зразків. До вибірки додаються кадри, що містять елементи телеметрії та приціли за повної відсутності безпілотника. Це розриває логічний ланцюжок між наявністю графіки та присутністю цілі. Завдяки класифікаційному компоненту функції втрат показник впевненості для зон з артефактами примусово знижується до рівня чистого фону.

Додатковим фільтром є безумовне вибракування кадрів, де масивні водяні знаки перекривають значну частину цілі. Будь-які спроби їх відновлення утворюють візуальні дефекти, які мережа може сприйняти як унікальні ознаки об'єкта.

Перспективним напрямком для подальшого дослідження є методологія оборонного донавчання (Defensive Fine-Tuning) у поєднанні з доменно-адверсаріальною адаптацією (DANN). Концептуально цей підхід передбачає навчання моделі таким чином, щоб вона не могла ідентифікувати джерело походження кадру (чисте бойове відео чи OSINT-матеріали з артефактами) – тобто ознаки, що вилучаються, стають інваріантними [22, с. 2]. Такий підхід теоретично дозволяє усунути доменний зсув безпосередньо у латентному просторі представлень. Проте через створення окремої архітектурної гілки та тривалого циклу балансування функції втрат цей напрям залишається предметом майбутніх досліджень.

Остаточна ефективність запропонованих методів буде оцінена на етапі практичного навчання моделей.

2.2.3 Методологія напівавтоматичного збору даних

Специфіка роботи з відкритими джерелами, зокрема відсутність структурованих метаданих та уніфікованих технічних описів, робить повноцінну автоматизацію збору даних неефективною. Це вимагає впровадження комбінованого підходу, де процес формування бази відеоматеріалів розділено на два етапи.

Перший етап полягає у моніторингу та екстракції даних із верифікованих OSINT-ресурсів підрозділів безпілотної авіації. На другому етапі здійснюється експертний візуальний контроль якості, під час якого відкидаються:

– відео з ракурсом «земля-повітря»;

– епізоди перехоплення нерелевантних для дослідження дронів;

– кадри з високим рівнем цифрового шуму;

– матеріали з критичним перекриттям цілі водяними знаками.

Саме верифіковані вручну відеоматеріали стають основою майбутнього датасету. Після проходження експертного контролю ці послідовності надходять до автоматизованого конвеєра для екстракції окремих кадрів та їхнього подальшого алгоритмічного очищення.

Після очищення від графічних артефактів відеоматеріали трансформуються у набір статичних зображень. Для формування якісного датасету реалізовано конвеєр попередньої обробки, що базується на наступних принципах:

– вибірка структурується відповідно до завдань: для навчання детектора виділяються окремі кадри, а для валідації системи супроводження зберігаються безперервні відеопослідовності. Це необхідно для коректної оцінки роботи фільтра Калмана та розрахунку метрик трекінгу;

– для запобігання перенавчанню моделі на ідентичних фонах проводиться фільтрація дублікатів. Це дозволяє залишити у вибірці лише кадри з унікальними ракурсами та зміщеннями об’єкта, що підвищує здатність мережі до узагальнення;

– на етапі підготовки даних виконується відбраковування технічного сміття: кадрів із критичним розмиттям, засвітами від вибухів або повною втратою сигналу. Такий фільтр гарантує, що модель навчатиметься на інформативних зразках;

– усі валідовані зображення зводяться до єдиного колірного простору та формату, сумісного з вимогами архітектур YOLO. Кожному кадру зіставляється файл анотацій, що забезпечує цілісність структури датасету та відсутність помилок під час ініціалізації навчання.

2.2.4 Методологія розмітки та стратегія монокласової детекції

Фундаментальною стратегією підготовки даних є відмова від розпізнавання конкретних типів БПЛА на користь використання єдиного класу «Target». З математичної точки зору такий підхід дозволяє концентрувати всю ймовірнісну масу на одному виході нейромережі, що суттєво підвищує повноту детекції та мінімізує кількість хибних пропусків. Це рішення також повністю усуває проблему міжкласового дисбалансу вибірки, оскільки на етапі термінального наведення пріоритетним є стабільне захоплення цілі, а не її ідентифікація.

Важливою умовою є збереження просторово-часової зв’язності відеопотоку. Використання темпорального контексту дозволяє точно ідентифікувати мікророзмірні об’єкти та зберігати сталий ідентифікатор Track ID при їх повторній появі в кадрі. Такий підхід є критичним для коректного розрахунку метрик якості супроводження, таких як MOTA та IDF1, і забезпечує надійну роботу контуру керування перехоплювачем.

Враховуючи значний обсяг вибірки та специфіку відеоданих, використання виключно ручної розмітки є неефективним. Для прискорення процесу реалізовано напівавтоматизований пайплайн на основі візуально-мовної моделі Meta Segment Anything Model 3.

Технологія передбачає два послідовні етапи. Спочатку за допомогою модуля Video Predictor здійснюється генерація псевдоанотацій. Модель ініціалізується на першому кадрі за допомогою текстового або візуального промту, після чого автономно сегментує об'єкт уздовж усієї послідовності. Результати експортуються у форматі Multiple Object Tracking, що дозволяє зберегти сталий Track ID для подальшого навчання алгоритмів супроводження. На другому етапі отримані дані проходять експертну валідацію в середовищі CVAT. Гібридний підхід, що поєднує автоматичну сегментацію та ручне коригування меж рамок, дозволяє експоненційно прискорити формування датасету без втрати якості розмітки.

Попри використання монокласової стратегії детекції, критично важливим аспектом є інтелектуальний аналіз структури даних для контролю репрезентативності вибірки. Кожна відеопослідовність класифікується за трьома доменами: типом цілі (Shahed, Орлан, Zala, Supercam), видом сенсора (RGB, NIR, IR) та ландшафтним фоном (чисте небо, поле, забудова, вода). Це необхідно для фіксації природного дисбалансу цілей на полі бою та запобігання перенавчанню мережі на специфічні умови фону, наприклад, лише на чисте небо.

Для автоматизації аналізу застосовано метод кодування метаданих безпосередньо у назвах файлів. Кожен кадр отримує ім'я за шаблоном: [Тип_цілі]_[Тип_камери]_[Фон]_[ID_відео]_[Номер_кадру].jpg.

Це дозволяє миттєво обчислювати статистику датасету за допомогою скриптів та ефективно фільтрувати дані перед етапом аугментації.

2.2.5 Теоретичне обґрунтування обмежень та відмови від стандартних стратегій аугментації

Аугментація в задачах комп'ютерного зору традиційно розглядається як інструмент компенсації статистичних дисбалансів вибірки. Проте для специфічних умов повітряного перехоплення малорозмірних БПЛА класичні геометричні, колірні та композиційні перетворення мають суттєві обмеження, що робить їх використання недоцільним або навіть шкідливим:

– недоцільність імітації апаратних перешкод. Теоретично, внесення штучного аналогового радіошуму розглядається як спосіб підвищення завадостійкості моделі. Проте аналіз апаратної архітектури перехоплювача спростовує цю необхідність. Оскільки бортовий обчислювач отримує чистий цифровий сигнал безпосередньо з оптичного сенсора через інтерфейси MIPI CSI-2 або USB, модель у реальному польоті працює з неспотвореним відеопотоком. Додаткове внесення штучного шуму, який виникає лише при передачі сигналу по радіоканалу до наземної станції,

викликало б небажаний штучний доменний зсув, змушуючи модель виділяти завади як корисні фонові ознаки;

– обмеження композиційних методів. Популярні методи змішування кадрів виключені з конвеєра навчання через високу ймовірність випадкового видалення або критичного пошкодження силуетів мікроцілей під час випадкового обрізання кадрів;

– синтез сцен. Вирізання силуетів рідкісних БПЛА та їх накладання на фони є перспективним інструментом подолання дисбалансу класів. Проте для збереження максимальної текстурної та фізичної автентичності оригінальних градієнтів на початковому етапі розробки цей підхід було вирішено відкласти на користь використання виключно реальних, немодифікованих кадрів бойової роботи;

З огляду на зазначені фактори, у межах цієї роботи було прийнято обґрунтоване рішення про відмову від використання стандартних просторово-колірних аугментацій та штучного зашумлення під час навчання моделей. Це гарантує, що нейромережа навчатиметься виключно на оригінальних, фізично достовірних градієнтах реального повітряного бою, запобігаючи деградації узагальнюючої здатності детектора.

2.2.6 Запобігання витоку даних та поділ вибірки

Коректний розподіл даних є критичним фактором для уникнення штучного завищення метрик та забезпечення об'єктивності результатів. Для досягнення повної статистичної незалежності вибірок поділ датасету здійснюється виключно на рівні цілісних оригінальних відеопослідовностей до моменту застосування будь-яких процедур аугментації.

Такий підхід дозволяє одночасно усунути три ключові ризики. По-перше, він виключає темпоральний витік інформації, що виникає при випадковому змішуванні суміжних кадрів, які мають майже ідентичний вміст. По-друге, це запобігає доменному витоку, оскільки всі фрагменти

одного епізоду перехоплення зі спільним фоном та артефактами залишаються в межах однієї групи. По-третє, це мінімізує ймовірність зміщення розподілу, гарантуючи, що складні умови навколишнього середовища будуть представлені в усіх частинах датасету рівномірно.

Після завершення підготовки матеріалу проводиться стратифікований розподіл відеопослідовностей у пропорції 70% для навчання та по 15% для валідації і фінального тестування. Групи формуються за допомогою алгоритму багатовимірної стратифікації, що забезпечує пропорційне представлення різних типів цілей, сенсорів та ландшафтних фонів.

Окремо для етапу цілочисельної квантизації у формат INT8 з навчальної вибірки виділяється калібрувальний набір обсягом 100–500 випадкових кадрів. Використання даних виключно з Train Split дозволяє точно обчислити динамічний діапазон активацій нейромережі, повністю виключаючи ризик прихованого витоку інформації з тестового набору. Усі процедури аугментації застосовуються лише до навчальної групи, тоді як тестова вибірка залишається незмінною. Це гарантує отримання достовірних метрик ефективності моделі під час роботи з реальним неспотвореним відеопотоком.

2.2.7 Метрики оцінки якості датасету

Перед початком навчання сформований масив даних проходить верифікацію для виявлення прихованих аномалій. Оцінка якості здійснюється за чотирма ключовими напрямками:

– дедуплікація та контроль витоку. Специфіка відкритих джерел передбачає велику кількість повторів одного й того самого контенту. Для виявлення побайтових копій використовується порівняння MD5-хешів, а для ідентифікації перекодованих або обрізаних версій відео – перцептивне хешування pHash. Це критично важливо для виключення витоку інформації між навчальною та тестовою вибірками;

– аналіз просторової консистентності. Для перевірки рівномірності розташування об'єктів будується теплова карта координат. Якщо цілі у вибірці зосереджені лише в центрі кадру, модель генеруватиме низьку впевненість на периферії. Верифікація через Bounding Box Heatmap гарантує, що система зможе стабільно детектувати дрон навіть під час його різкого виходу з центру поля зору камери;

– перевірка темпоральної цілісності та хибно-порожньої анотації. На цьому етапі перевіряється узгодженість Track ID. Головною загрозою напівавтоматичної розмітки за допомогою SAM 3 є раптова втрата об'єкта алгоритмом. Якщо видимий у кадрі БПЛА не отримує рамку і потрапляє у вибірку як фон, мережа сприймає це як сигнал ігнорувати ціль. Це призводить до суперечливого навчання та різкого падіння показника повноти, тому відсоток таких помилок суворо контролюється;

– перевірка доменного балансу. Фінальний етап передбачає порівняння частоти появи типів цілей, фонів та умов освітлення між усіма групами даних. Візуальний аналіз через гістограми дозволяє підтвердити, що тестова вибірка відповідає складності навчальної.

2.2.8 Обґрунтування цільового обсягу вибірки

Визначення оптимального обсягу вибірки базується на співвідношенні кількості параметрів моделі та інформаційної ємності датасету. Оскільки цільова архітектура YOLO26n містить близько 2,5 млн параметрів, цільовий обсяг навчального набору у ~7000 позитивних кадрів є науково обґрунтованим.

Дослідження Liu & Juang (2026) на датасеті VisDrone (6471 тренувальне зображення) продемонструвало успішну оптимізацію ваг моделі YOLOv8l (43.6 млн параметрів) [23]. Для моделей класу Nano, які мають у 4–14 разів менше параметрів, обсяг у ~7000 кадрів є гарантовано достатнім для збіжності алгоритму без ризику недонавчання.

Необхідний обсяг та різноманітність даних забезпечуються шляхом масштабування кількості залучених унікальних OSINT-відеоматеріалів реальної бойової роботи та їхнього ретельного покадрового відбору.

Підсумовуючи теоретичне обґрунтування методології формування датасету, можна зробити такі ключові висновки:

– аналіз підтвердив непридатність публічних датасетів через відсутність ракурсу повітря-повітря та релевантних військових об’єктів. Це робить формування власної бази на основі OSINT-матеріалів єдиним інженерно виправданим рішенням;

– цільовий обсяг тренувального набору визначено на рівні 7 000 кадрів, що є оптимальним порогом для конвергенції моделі класу Nano;

– впроваджений конвеєр підготовки даних, що поєднує очищення артефактів та напівавтоматичну розмітку моделлю SAM 3, дозволяє експоненційно прискорити підготовку вибірки. Збереження анотацій у форматі MOT забезпечує темпоральну зв’язність, необхідну для навчання алгоритмів трекінгу;

– застосування стратифікованого поділу вибірок у пропорції 70:15:15 на рівні відеоепізодів разом із методами цифрової дедуплікації виключає ризик витоку інформації. Це забезпечує об’єктивність оцінки роботи моделі на реальному відеопотоці.

2.3 Дослідження методів оптимізації моделей глибокого навчання для Edge-пристроїв

Донавчання моделі детекції є лише початковим етапом розробки. За замовчуванням ваги нейромережі експортуються у форматі 32-бітної плаваючої коми, пряме використання якого на бортових обчислювачах є неможливим. Запуск такої моделі в умовах обмежених ресурсів створює обчислювальне перевантаження, що призводить до критичного падіння частоти кадрів, перегріву чипа та швидкого виснаження батареї.

Для досягнення цільової швидкодії та стабільної роботи алгоритмів супроводження нейромережа потребує глибокої структурної та форматної трансформації. З метою балансу між точністю та енергоефективністю у роботі застосовується комплексний конвеєр оптимізації, що складається з архітектурного прунінгу, зниження розрядності ваг через квантизацію та низькорівневої компіляції обчислювального графа.

2.3.1 Прунінг

Першим етапом оптимізації є прунінг – алгоритмічне вилучення ваг, що мають найменший вплив на результат. З двох існуючих підходів для бортових прискорювачів застосовується виключно структурний прунінг на основі величини ваг. Неструктурний підхід перетворює матриці на розріджені, обчислення яких тензорні ядра не здатні прискорити через оптимізацію під щільне множення.

Структурний метод вилучає цілі згорткові фільтри або канали. Це дозволяє фізично зменшити розмірність моделі та гарантовано знизити кількість операцій. Жорсткою інженерною вимогою при цьому є дотримання апаратної кратності: кількість залишених фільтрів у кожному шарі має бути кратною 16 або 32. Прунінг до довільних значень призводить до неефективного використання регістрів процесора і парадоксального уповільнення роботи моделі.

На відміну від важких моделей, де можна безболісно вилучити до половини параметрів [24, с. 10, 20], застосування прунінгу до ультралегких Edge-архітектур має суттєві обмеження. Моделі класу Nano мають мінімальну кількість параметрів, що фактично є нижньою межею інформаційної ємності для архітектур такого типу. Через низьку структурну надмірність цих нейромереж прунінг є вкрай ризикованим етапом, оскільки навіть незначне видалення фільтрів може спричинити колапс точності детектора.

З огляду на ці фундаментальні обмеження прунінгу для легких нейромереж, у межах цієї роботи було прийнято рішення відмовитися від практичного застосування прунінгу, а основне навантаження щодо прискорення інференсу покласти на квантизацію ваг та оптимізацію обчислювального графа.

2.3.2 Квантизація

Оскільки агресивний прунінг для моделей класу Nano недоцільний, головним методом апаратного прискорення виступає квантизація після навчання (Post-Training Quantization). Цей процес передбачає конвертацію ваг моделі з формату 32-бітної плаваючої коми у значення нижчої розрядності. Такий перехід радикально зменшує споживання оперативної пам'яті, розвантажує шину даних і дозволяє задіяти спеціалізовані тензорні ядра.

Стратегія квантизації у роботі базується на кількох підходах залежно від цільової розрядності. Використання формату напівточності FP16 є базовим рівнем для легких моделей. Його динамічний діапазон дозволяє уникнути спотворення дрібних ознак об’єкта. Завдяки компіляції перехід на FP16 забезпечує фундаментальне прискорення інференсу у 3–5 разів порівняно з базовим середовищем PyTorch, практично без втрати mAP [25, с. 1].

Більш агресивним методом є перехід до цілочисельної розрядності INT8. Для легких оптимізованих архітектур деградація точності за такого підходу зазвичай не перевищує 5,6% mAP, тоді як для важких моделей цей показник може сягати 23% [26, с. 1]. Для мінімізації втрат використовується калібрувальний датасет обсягом до 500 репрезентативних кадрів. Це дозволяє обчислити точні коефіцієнти масштабування та коректно перекласти ваги у діапазон від –128 до 127.

Компромісним рішенням виступає змішана квантизація. У цьому разі найбільш чутливі шари, зокрема вихідні «голови» детектора, залишаються у форматі FP16, а масивні внутрішні блоки згорток конвертуються в INT8. Застосування цих методів оптимізації є ключовою умовою для усунення обчислювальних заторів та досягнення цільової швидкодії понад 30 кадрів за секунду на бортових обчислювачах.

2.3.3 Компіляція та апаратне прискорення

Завершальним етапом підготовки є трансформація моделі з формату ONNX у бінарний рушій інференсу. Головним методом тут виступає графова оптимізація, зокрема злиття шарів, що об'єднує операції згортки та активації в єдину інструкцію. Це мінімізує переміщення даних між регістрами та пам’яттю, що є критичним для ресурсів Edge-пристроїв.

Еталонним інструментом компіляції обрано NVIDIA TensorRT. Попереднє оцінювання швидкодії та затримки моделі після компіляції здійснюється утилітою симуляції trtexec. Попри обмеження прямої екстраполяції результатів на вбудовані платформи, симуляція дозволяє підтвердити сумісність моделі з TensorRT та оцінити відносний приріст її швидкодії.

Для досягнення швидкодії понад 30 FPS необхідно поєднувати структурну трансформацію моделі з компіляцією графа. Через низьку надмірність архітектур класу Nano пріоритетним шляхом прискорення є не прунінг, а квантизація ваг.

Використання екосистеми TensorRT як еталонного середовища дозволяє оцінити потенціал оптимізації обчислювального графа та підтвердити теоретичну відповідність моделі обмеженням цільового сегмента. Такий підхід гарантує, що оптимізована модель відповідатиме обмеженням масового сегмента засобів перехоплення, зберігаючи при цьому баланс між швидкістю інференсу та повнотою детекції.

3 МЕТОДОЛОГІЯ ПІДГОТОВКИ ДАНИХ

3.1 Методологія збору та джерела первинних OSINT-даних

Для розробки надійної нейромережевої системи детекції БПЛА в реальних умовах повітряного перехоплення критично важливим етапом є формування репрезентативного навчального набору даних. У межах цієї роботи як єдине джерело первинної інформації було обрано дані з відкритих джерел, зокрема публічні Telegram-канали українських військових підрозділів.

Обґрунтування доцільності використання саме OSINT-джерел базується на необхідності повного подолання доменного зсуву, який виникає при спробах застосування синтетичних або стандартних публічних датасетів. Традиційні набори даних комп'ютерного зору, такі як VisDrone2019, Anti-UAV або DroneVSBird, детальний аналіз яких наведено у розділі 2 (таблиця 2.2), характеризуються статичною наземною перспективою спостереження та фокусуються переважно на цивільних мультикоптерах комерційного класу. У бойових реаліях дрон-перехоплювач працює в динамічній перспективі «повітря–повітря», де цілями виступають військові БПЛА літакового типу, як-от «Орлан-10», «Zala», «Supercam» чи баражуючі боєприпаси сімейства «Shahed».

Генерація синтетичних даних за допомогою 3D-симуляторів була повністю відхилена як з технічних, так і з практичних міркувань. По-перше, у відкритому доступі відсутні спеціалізовані симулятори для дронів-перехоплювачів, які б дозволили змоделювати кінематику та перспективу повітряного бою. По-друге, наявні загальні симулятори безпілотників використовують спрощений ігровий графічний рушій. Згенеровані в них зображення мають характерний синтетичний вигляд, який за своєю деталізацією, текстурами та поведінкою світла значно відрізняється від реального світу. Крім того, подібні симуляції пропонують дуже обмежений

набір типових сценаріїв польоту та умов оточення. Спроба навчити нейромережу на таких спрощених ігрових кадрах призвела б до критичного доменного зсуву, роблячи модель абсолютно нездатною розпізнавати об'єкти під час перших випробувань на реальному перехоплювачі. Таким чином, OSINT-відеоматеріали реальної бойової роботи є єдиним джерелом отримання достовірних сигнатур повітряних цілей на тлі автентичної підстильної поверхні та за природних умов освітлення й кутів зйомки.

Під час аналізу специфіки OSINT-матеріалів було виявлено, що практично всі доступні відеозаписи містять елементи OSD, тобто накладену графіку польотної телеметрії, як-от прицільні марки, координати, рівень заряду батареї, параметри сигналу зв'язку. В межах цього дослідження встановлено, що OSD-елементи є невід'ємною частиною робочого домену системи, і їхня наявність у навчальній вибірці є технічно неминучою, оскільки польотна телеметрія накладається безпосередньо бортовим мікропроцесором на відеосигнал перед його передачею по радіоканалу на наземну станцію оператора, де й відбувається фіксація відеозапису.

Натомість апаратна архітектура БПЛА-перехоплювача з інтегрованою системою штучного інтелекту функціонує за принципом розділення відеопотоків. Бортовий обчислювач, як правило, зчитує первинний чистий цифровий сигнал безпосередньо з матриці сенсора камери через інтерфейси MIPI CSI або USB. Обробка кадрів нейромережею та алгоритмами трекінгу відбувається на оригінальних зображеннях до будь-якої апаратної чи програмної обробки польотним контролером, який накладає сервісну графіку для оператора паралельно або на наступних етапах конвеєра обробки та передачі сигналу. У режимі автономного наведення на ціль потреба у візуалізації польотної телеметрії для людини втрачає актуальність, оскільки контур керування об'єктом замикається бортовими алгоритмами автоматичного супроводження.

Така невідповідність створює доменний зсув між навчальною вибіркою та реальним операційним середовищем. Через притаманну

нейромережам тенденцію до вибору найпростіших ознак модель під час навчання на OSINT-матеріалах намагатиметься побудувати хибну просторову кореляцію між OSD та наявністю БПЛА, що впливатиме на здатність моделі ефективно визначати ціль в умовах бойового застосування.

Окрім наявності OSD, існує ще одне джерело доменного зсуву – це водяні знаки та логотипи військових підрозділів або благодійних організацій, накладені вже після завершення польоту під час публікації матеріалів. Оскільки ці графічні артефакти мають високий контраст та стабільне положення в кутах кадру, вони створюють критичну загрозу навчання за спрощеними шляхами, коли модель починає асоціювати статичний логотип із присутністю цілі. Для подолання цього явища у роботі обґрунтовано необхідність інтеграції спеціалізованих негативних зразків, що містять водяні знаки та OSD за повної відсутності цільових об'єктів. Це дозволяє примусово зруйнувати логічний зв'язок між наявністю накладених інтерфейсів і присутністю цілі, змушуючи глибокі шари моделі орієнтуватися виключно на морфологічні ознаки безпілотника.

3.2 Первинна фільтрація та критерії вибракування вихідних файлів

Використання відеоматеріалів, отриманих із відкритих джерел, супроводжується значною кількістю специфічних завад, притаманних військовому фільмуванню реальних бойових дій. Для формування якісного набору даних та запобігання деградації архітектури детектора було розроблено та застосовано систему суворих критеріїв первинної фільтрації вихідних медіафайлів.

У межах дослідження було проаналізовано та вручну відібрано матеріали з мілітарних Telegram-каналів безпосередніх виробників та операторів БПЛА (зокрема «Дикі Шершні», «Signum»), а також офіційних медіаресурсів волонтерських організацій та благодійних фондів (Фонд

Сергія Притули, Спільнота Стерненка, «Повернись живим»). Початковий набір накопичених матеріалів становив 154 відеофайли.

Під час експертного візуального аналізу було вибракувано та повністю виключено з подальшої обробки 36 відеофайлів через невідповідність таким критеріям якості:

– критичне перекриття цілі графічними OSD-елементами. Матеріали, у яких малі малорозмірні БПЛА-цілі на критичних дистанціях виявлення повністю або частково перекривалися великими прицільними марками, динамічними рамками захоплення або штучно розмитими географічними координатами у кутах кадру, були визнані непридатними;

– наявність штучних геометричних меж кадру. Частина відеоматеріалів містила вертикальні або горизонтальні чорні смуги, що з'являються внаслідок невідповідності оригінального формату запису стандартному співвідношенню сторін екрана. Такі відео були безумовно відхилені. Згідно з дослідженнями упередженості нейромереж до простоти та ефекту «Розумного Ганса», згорткові шари схильні використовувати контрастні статичні межі кадру як помилкові просторові орієнтири для виявлення об'єктів. Оскільки в реальних бойових умовах на борту перехоплювача таких штучних меж не існує, їх наявність у вибірці призводить до колапсу узагальнюючої здатності моделі при роботі з чистим відеопотоком;

– аналогові та радіочастотні перешкоди. Особливу увагу було приділено фільтрації відеоматеріалів із високим рівнем аналогового шуму та інтерференційних завад. Попри поширену гіпотезу про доцільність навчання нейромережевих моделей на спотворених завадами РЕБ або згасанням сигналу зображеннях, аналіз архітектурних особливостей бортового комплексу спростовує цю необхідність. Відеоматеріали з відкритих джерел фіксуються наземною станцією керування, куди відеопотік транслюється бездротовим радіоканалом, вразливим до затухання сигналу, обмежень зони прямої видимості та деструктивного

впливу засобів радіоелектронної боротьби. Натомість програмний комплекс автономного наведення функціонує безпосередньо на бортовому обчислювачі БПЛА-перехоплювача. Передача відеосигналу від оптичного сенсора до процесора обробки даних здійснюється через локальний апаратний інтерфейс (наприклад, MIPI CSI-2 або USB) безпосередньо в межах бортової системи, що повністю виключає вплив електромагнітних завад бездротового каналу зв'язку на якість вхідного відеопотоку. Таким чином, у реальному контурі інференсу на борту БПЛА модель отримує неспотворений первинний відеосигнал. Використання для навчання кадрів із високим рівнем аналогового радіошуму спричиняє додатковий штучний доменний зсув та призводить до виділення моделлю завад як корисних фонових ознак, що критично знижує узагальнюючу здатність класифікатора в реальних умовах експлуатації. З огляду на це, відеоматеріали з вираженими аналоговими перешкодами підлягали безумовному виключенню з вибірки на етапі первинного аналізу та подальшої покадрової анотації.

У результаті застосування розробленої системи багатокритеріальної фільтрації з початкового масиву у 154 відеозаписи було сформовано верифіковану вихідну вибірку зі 118 відеопослідовностей, параметри яких відповідають кінематичним та оптичним обмеженням функціонування бортових систем автономного супроводження цілей у реальному часі.

3.3 Ручна сегментація та виділення поодиноких епізодів збиття

Первинні відеоматеріали, отримані з OSINT-джерел, переважно є відеокомпіляціями, що складаються з кількох епізодів ураження цілей. Для подальшої обробки виникла необхідність проведення точної часової сегментації з метою виділення індивідуальних епізодів, у межах яких фігурує лише один об'єкт ураження. З метою мінімізації часових витрат та запобігання деградації вихідної якості відеопотоку в процесі редагування

було обрано спеціалізоване програмне забезпечення LosslessCut. Ключовою перевагою цього інструменту є можливість безвтратного обрізання відео шляхом безпосереднього копіювання бітового потоку без перекодування, що повністю зберігає оригінальну роздільну здатність та артефакти стиснення вихідного матеріалу.

Для систематизації процесу обробки та запобігання надлишковим операціям було розроблено структуровану схему каталогізації файлів. Кожне вихідне багатоепізодне відео завантажувалося у середовище редагування під своїм унікальним ідентифікатором. Виділені з нього поодинокі сегменти експортувалися в окрему директорію, назва якої повністю відповідала імені вихідного файлу-компіляції. Такий ієрархічний підхід дозволив чітко розмежувати вже оброблені відеоматеріали від тих, що лише очікували на сегментацію, унеможливлюючи повторну обробку однакових даних та оптимізуючи загальні витрати часу.

Важливою вимогою при виборі програмного інструменту була підтримка пакетної обробки за один робочий цикл. Функціонал LosslessCut дозволив розставляти маркери початку та кінця для всіх корисних епізодів у межах одного файлу одночасно, після чого виконувався їхній одноразовий експорт. Це виключило необхідність багаторазового відкриття одного й того самого відеофайлу, знизило ризик втрати поточної позиції відтворення та суттєво підвищило продуктивність роботи.

У результаті описаного процесу часової сегментації було отримано 610 поодиноких епізодів ураження цілей. Початкова кількість потенційних епізодів у компіляціях була значно більшою, проте частина з них була відсіяна безпосередньо на етапі поділу. Основними причинами відбраковування на цьому етапі стали невідповідність критеріям візуальної якості, а також надмала тривалість епізодів.

3.4 Автоматизована класифікація метаданих за допомогою VLM

Для проведення детального статистичного аналізу набору даних, а також для реалізації методологічно коректного стратифікованого розподілу вибірки на тренувальну, валідаційну та тестову частини, виникла необхідність класифікації зібраних відеоматеріалів. З огляду на значний обсяг вихідних даних, повністю ручне розмічення є неефективним і часомістким.

Для вирішення цієї задачі було розроблено та реалізовано автоматизований пайплайн семантичного маркування відеопотоку на основі мультимодальної мовної моделі Gemini 2.5 Pro через прикладний програмний інтерфейс Vertex AI API (програмна реалізація наведена у додатку А).

Передача повного відеопотоку високої роздільної здатності до хмарного API супроводжується значними обчислювальними витратами та обмеженнями за обсягом вхідного контексту. Для оптимізації процесу було розроблено алгоритм динамічного відбору ключових кадрів, інтегрований у загальний конвеєр обробки. Його логіка побудована на динамічному визначення часового діапазону та фільтрації за рівне різкості.

Динамічне визначення часового діапазону є алгоритмом, який обчислює загальну тривалість відео на основі частоти кадрів та їхньої загальної кількості у файлі. Кількість цільових кадрів для аналізу динамічно масштабується в межах від 3 до 15 залежно від хронометражу. Для відео тривалістю понад 15 секунд екстракція кандидатів виконується виключно з термінальної фази польоту, тобто від 70% до 100% загального часу відео. Це обумовлено тим, що на фінальній стадії зближення БПЛА-ціль розташована найближче до камери перехоплювача, має найбільший кутовий розмір і піддається кращій детекції. Для відео коротших за 15 секунд діапазон вибірки становить від 10% до 90% часової шкали, виключаючи початкові та кінцеві кадри через можливі малі розміри цілі на

початку відео або максимальне зближення з фюзеляжем цілі на останніх кадрах, що унеможливлює класифікацію цілі.

Фільтрація за рівнем різкості використовується, шоб виключити з аналізу кадри з кінематичним розмиттям або цифровими завадами, алгоритм вираховує оцінку різкості для кожного кадру-кандидата через дисперсію фільтра Лапласа. Для аналізу моделлю VLM відбираються лише найбільш різкі та інформативні кадри, які згодом сортуються в оригінальному хронологічному порядку та трансформуються в колірний простір RGB. Сформований пакет зображень передається моделі gemini-2.5-pro разом із системним промтом.

Промт розроблено з урахуванням специфіки бойового відеопотоку та містить правила запобігання помилкам, спричиненим наявністю OSD-інтерфейсу та вотермарок:

– ігнорування штучних маркерів. Моделі заборонено класифікувати приціли у центрі екрана як хрестоподібну форму дрона;

– ігнорування вотермарок. Статичні логотипи каналів або значки у формі дронів на екрані не повинні сприйматися як реальні повітряні цілі.

Для забезпечення стабільної автоматизованої обробки результатів, запит до моделі було спроєктовано таким чином, щоб відповідь поверталася у строгому форматі JSON із трьома основними полями:

а) геометрична форма цілі:

– delta – дельтоподібне або трикутне крило, стрілоподібний планер без вираженого трубчастого фюзеляжу;

– wing – класичне літакове крило з вираженим подовженим розмахом та тонкішим фюзеляжем;

– cross – хрестоподібний силует із характерними подвійними X-подібними крилами;

– no_drone – негативні кадри, які не містять фізичного об'єкта в небі (лише завади, OSD або порожнє небо), необхідні для запобігання ефекту shortcut learning;

б) тип сенсора камери:

– rgb – денний кольоровий спектр із природним відображенням кольорів та тіней;

– thermal – інфрачервоний тепловізійний спектр, що характеризується монохромним висококонтрастним відображенням гарячих силуетів на однорідному тлі;

в) тип підстильної поверхні:

– sky – чисте небо або хмари без ліній рельєфу;

– field – земна поверхня під лінією горизонту з патерном полів або доріг;

– horizon – змішане тло, де лінія горизонту перетинає силует цілі;

– complex – складні антропогенні або природні структури (ліс forest, міська забудова urban, вода uniform), об'єднані в один клас для підвищення щільності вибірки.

Після успішного інференсу та отримання структурованої відповіді JSON для кожного відеоепізоду, розроблений Python-скрипт вилучав семантичні теги та динамічно формував нову назву для всього відеофайлу за шаблоном: [форма_цілі]_[тип_камери]_[тип_тла]_[номер_відео].[розширення].

Попри високу точність моделі Gemini 2.5 Pro, наявність граничних умов, наприклад, складні переходи на межі класів sky та horizon, або тепловізійні кадри з високим шумом, вимагала залучення людини в контур управління для фінальної перевірки. Було проведено ручний аудит результатів класифікації, під час якого виправлено похибки у визначенні суміжних категорій фонів та сенсорів.

Такий підхід дозволив закодувати метадані безпосередньо у фізичних назвах файлів відеоепізодів. Це значно спростило подальший автоматизований статистичний аналіз та формування стратифікованих

вибірок, оскільки будь-який програмний модуль міг миттєво зчитати семантичні ознаки епізоду за допомогою регулярних виразів.

3.5 Повторна фільтрація крос-джерельних дублікатів відеоепізодів

Під час перевірки результатів класифікації відео було виявлено 3 ідентичні відеозаписи збиття БПЛА, які були успішно видалені вручну. Цей факт вказав на наявність характерного дефекту вхідних OSINT-матеріалів, спричиненого людським фактором: один і той самий військовий підрозділ під час монтажу різних звітних нарізок міг випадково повторно використати той самий фрагмент успішного перехоплення у різних роликах. Як наслідок, під час сегментації вихідних 118 компіляцій на поодинокі епізоди утворилися дубльовані ролики.

Наявність таких дублікатів у наборі даних є критичною загрозою, яка може призвести до двох негативних наслідків:

– витоку даних, оскільки ідентичні відеоепізоди можуть одночасно потрапити як до тренувальної, так і до тестової вибірок, штучно завищуючи оціночні метрики точності моделі;

– спотворення розрахованих показників розподілу категорій під час аналізу набору даних.

Для автоматизації процесу очищення було проведено дедуплікацію відео за допомогою розробленого Python-скрипту перцептивної дедуплікації. Його запуск було реалізовано безпосередньо після етапу класифікації метаданих VLM, але до етапу статистичного аналізу, що гарантувало чистоту розрахованих метрик.

Для розв'язання цієї проблеми розроблений алгоритм базується на технології перцептивного хешування. Процес дедуплікації починається з формування візуальної сигнатури відео. Зчитування кожного кадру та обчислення його хешу є обчислювально надлишковою операцією, яка створює надмірне навантаження на підсистему пам'яті. З метою оптимізації

швидкодії алгоритм виконує тимчасове проріджування відеопотоку, зчитуючи рівно один кадр на секунду. Для кожного відібраного кадру OpenCV-тензор у форматі BGR трансформується у формат RGB та аналізується бібліотекою imagehash. Алгоритм pHash стискає зображення, застосовує дискретне косинусне перетворення, виділяє низькі частоти, що представляють структуру та градієнти кадру, і формує стійкий 64-бітний бінарний відбиток. Послідовність цих відбитків у результаті формує унікальну візуальну сигнатуру конкретного відео.

Наступним кроком є обчислення відстані Геммінга та порівняння отриманих сигнатур. За допомогою генератора унікальних пар itertools.combinations алгоритм проводить перехресне порівняння всіх сигнатур відео між собою за формулою комбінацій без повторень:

$$C_n^2 = \frac{n(n-1)}{2}, \tag{3.1}$$

де C_n^2 – кількість комбінацій без повторень (кількість унікальних пар відео для попарного порівняння);

n – загальна кількість унікальних відеоепізодів у базі даних.

Для кожної унікальної пари відео А, відео Б алгоритм попарно порівнює кожен кадр-хеш h_A з кожним кадром-хешем h_B. Мірою схожості виступає відстань Геммінга – кількість відмінних бітів між двома 64-бітними послідовностями:

$$D_H(h_A, h_B) \leq \theta_{hash}, \tag{3.2}$$

де D_H – відстань Геммінга між двома кадр-хешами;

h_A, h_B – 64-бітні бінарні відбитки порівнюваних відео;

θ_{hash} – поріг схожості кадрів для визначення дублікатів.

У скрипті встановлено жорсткий поріг $\theta_{\text{hash}} = 5$, що відповідає майже ідентичній візуальній структурі кадрів, але допускає дрібні зміни. Якщо хоча б для однієї пари кадрів виконується ця умова, Відео Б маркується як дублікат і заноситься до списку виключення.

За допомогою розробленого дедуплікатора було проаналізовано 610 відеоепізодів. Алгоритм успішно виявив та ізолював 13 потенційних дублікатів.

Було проведено ручну перевірку всіх 13 ізольованих файлів. У результаті перевірки було підтверджено 100% точність роботи алгоритму: всі 13 файлів виявилися крос-джерельними копіями вже наявних у базі відео. Усі 13 дубльованих відеоепізодів були остаточно видалені з набору даних.

Таким чином, фінальний обсяг очищеного та верифікованого набору унікальних відеоепізодів склав 594 відео (610 початкових епізодів за мінусом 16 вилучених дублікатів, 3 з яких було знайдено власноруч на етапі перевірки результатів класифікації VLM), що гарантувало повну відсутність витоку інформації та спотворення статистичного розподілу перед наступними етапами проєктування системи.

3.6 Аналіз характеристик та структурного розподілу набору даних

Після завершення етапів первинного відбору, ручної сегментації, автоматизованої класифікації та видалення виявлених дублікатів було отримано чисту, верифіковану базу унікальних відеоепізодів. На основі метаданих, закодованих у назвах файлів, було проведено детальний аналіз структури отриманих даних для планування наступних кроків проєктування системи.

За допомогою методів аналізу метаданих було зібрано статистичні характеристики часових параметрів усього набору даних:

– загальна кількість відео: 594 епізоди;

– сумарна тривалість усіх відео: 38 хвилин 33 секунди (2313.4 секунди);

– середня тривалість одного відеоепізоду: 3.9 секунди.

Аналіз сумарного хронометражу дозволив оцінити загальну щільність кадрового простору. При стандартній частоті запису 30 кадрів за секунду загальний обсяг вихідного матеріалу становить 69402 кадри. З математичної точки зору використання абсолютно всіх послідовних кадрів для навчання моделі YOLO є нераціональним та шкідливим через два ключові чинники. З одного боку, існує суттєвий ризик перенавчання моделі, оскільки сусідні кадри з інтервалом близько 33 мілісекунди є майже ідентичними. Навчання на такій надлишковій вибірці змусить нейромережу запам'ятовувати конкретні фони замість того, щоб узагальнювати унікальні ознаки цілей. З іншого боку, виникають критичні ресурсні обмеження, адже безпосереднє навчання на масиві з 69 тисяч кадрів призведе до надмірного збільшення тривалості процесу навчання та перевищення наявних обчислювальних лімітів.

Тому цей аналіз обґрунтував необхідність обов'язкового застосування методів кадрового прорідження.

Крім того, дослідження середньої тривалості одного відео та побудованої гістограми розподілу (рисунок 3.1) виявили високу варіативність: у вибірці присутні як вкрай короткі епізоди, так і тривалі послідовності. Це диктує жорстку вимогу до методології поділу даних на тренувальну, валідаційну та тестову вибірки. Поділ має бути стратифікованим не лише за семантичними класами цілей, а й за тривалістю відео. Нехтування фактором тривалості епізодів під час поділу набору даних створює суттєві ризики розбалансування темпоральної структури між вибірками. Випадковий розподіл без урахування довжини відеоматеріалів може призвести до критичної асиметрії у розподілі дисперсії та обсягу даних між групами навчання, валідації та тестування. Накопичення тривалих епізодів в одній із вибірок штучно перевантажить її

великою кількістю кадрів із низькою візуальною варіативністю через високу кореляцію суміжних кадрів у межах однієї послідовності, тоді як інша вибірка може отримати недостатній обсяг кадрів за надмірно високої різноманітності сюжетів. Такий дисбаланс варіативності даних не лише негативно впливає на процес оптимізації та збіжності моделі під час навчання, а й унеможливлює проведення репрезентативної та об'єктивної оцінки реальної точності детектора на етапі фінального тестування.

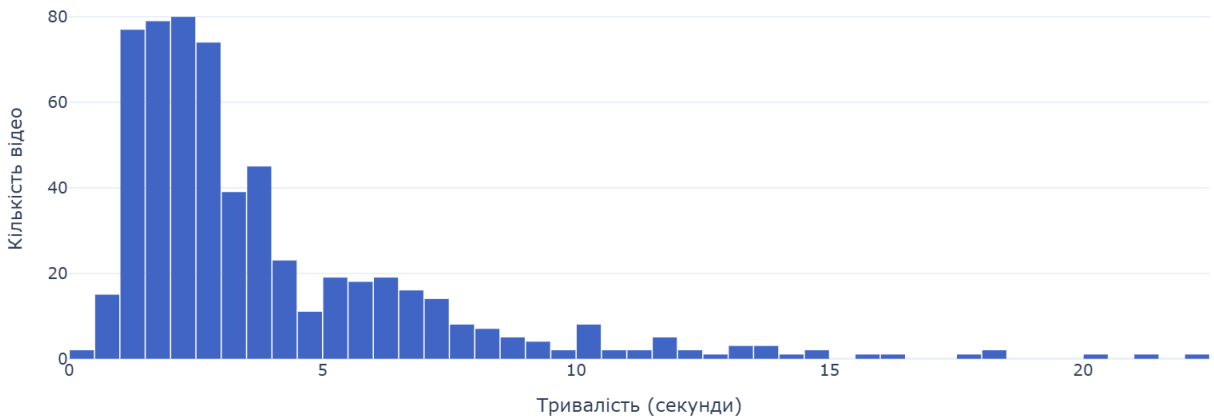
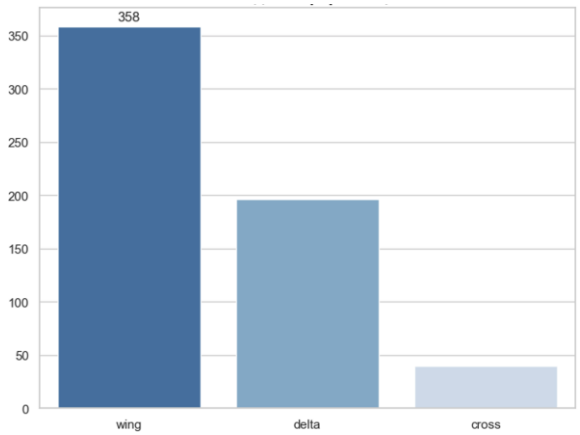


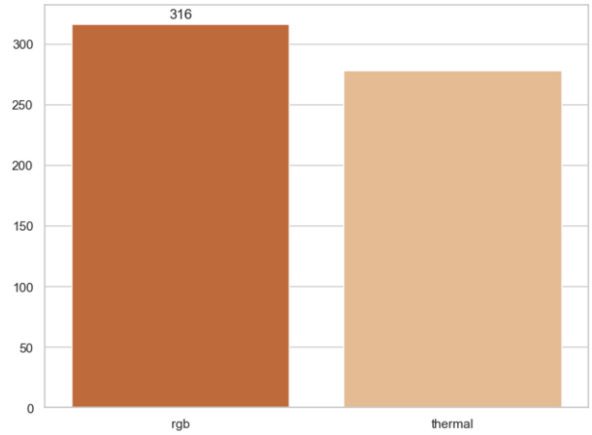
Рисунок 3.1 – Розподіл тривалості відео у наборі даних

На основі метаданих класифікації файлів було побудовано гістограми розподілу унікальних відео за трьома основними доменами метаданих (рисунок 3.2).

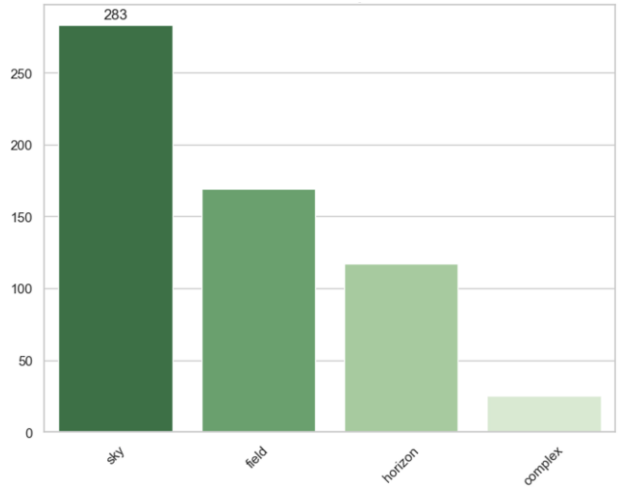
Як демонструють результати аналізу, у сформованому наборі даних спостерігається нерівномірний розподіл за всіма трьома досліджуваними ознаками. Зокрема, у категорії геометричної форми цілі домінує клас wing (класичне літакове крило), на який припадає 358 відеоепізодів, що становить 60,27% від загальної кількості. Водночас на клас delta (дельтоподібне крило) припадає 196 епізодів (32,99%), а найрідкіснішим є клас cross (хрестоподібний силует), представлений лише 40 відеоепізодами, що становить 6,74% (рисунок 3.2, а).



а) за формою цілі



б) за типом камери



в) за типом фону

Рисунок 3.2 – Розподіл унікальних відеоепізодів у наборі даних за категоріями метаданих: а) за формою цілі; б) за типом камери; в) за типом фону

У розподілі за типом сенсора спостерігається відносна рівномірність: оптичний спектр rgb охоплює 316 епізодів (53,20%), а тепловізійний thermal – 278 (46,80%) (рисунок 3.2, б). Таке збалансоване співвідношення забезпечує високу стійкість розроблюваної системи до зміни умов зовнішнього освітлення та часу доби.

Щодо типу підстильної поверхні, то більшість цілей спостерігається на тлі чистого неба або хмар (sky), що становить 283 епізоди (47,64%). На фоні земної поверхні чи полів (field) зафіксовано 169 епізодів (28,45%), на

лінії горизонту (horizon) – 117 епізодів (19,70%), а на тлі складного ландшафту (complex) – лише 25 епізодів, що становить 4,21% від загального обсягу вибірки (рисунок 3.2, в).

Отриманий статистичний дисбаланс класів є природним відображенням умов реального повітряного перехоплення і є повністю обґрунтованим у межах цієї роботи з двох основних причин.

Перша причина полягає в реальності бойового домену, оскільки наведена статистика відображає фактичну частоту появи та ураження засобів повітряного нападу противника в зоні бойових дій. Розвідувальні БПЛА літакового типу, клас wing, такі як «Орлан-10» або «Supercam», застосовуються та збиваються значно частіше, ніж ударні дрони-камікадзе з Х-подібним крилом, клас cross, такі як «Lancet». Аналогічно, більшість повітряних перехоплень зенітними засобами чи ракетами відбувається у верхній півсфері на тлі чистого неба або хмар, а не безпосередньо над складною підстильною поверхнею лісових масивів чи міської забудови.

Другою причиною є обмеженість відкритих джерел, адже для формування бази даних було залучено та проаналізовано фактично всі доступні у відкритому доступі якісні відеоматеріали повітряного ураження безпілотних апаратів. Будь-які спроби штучного вирівнювання класів шляхом випадкового видалення надлишкових зразків з домінуючих категорій призвели б до критичного та невиправданого зменшення загальної ємності вибірки. З огляду на це, отриманий обсяг у 594 унікальні відео визнано фінальним та оптимальним для подальшого проєктування системи.

Отримані статистичні закономірності довели об'єктивність наявного дисбалансу класів як прямого відображення реальних умов повітряного перехоплення, виключивши потребу у штучному скороченні домінуючих категорій. Разом з цим, вивчення параметрів відеопотоку обґрунтувало дві ключові вимоги до наступних етапів підготовки даних: по-перше, критичну необхідність впровадження алгоритмів дедуплікації для очищення часової надликовості та зменшення обчислювального навантаження; по-друге,

обов'язкове застосування багатовимірного стратифікованого розподілу вибірок із компенсацією впливу тривалості відеоепізодів для уникнення витоку інформації та забезпечення репрезентативності подальшої оцінки точності проєктованого детектора.

3.7 Методологія стратифікованого розподілу вибірки та запобігання витоку даних

Поділ набору даних на навчальну, валідаційну та тестову вибірки у пропорції 70:15:15 відповідно є визначальним кроком підготовки даних. Для забезпечення об'єктивності оцінок точності та стійкості моделей методологія розподілу у цій роботі базується на двох фундаментальних принципах.

Перший принцип вимагає запобігання витоку даних. Для цього розподіл здійснюється виключно на рівні цілісних відеоепізодів: усі кадри однієї траєкторії збиття БПЛА гарантовано потрапляють до однієї вибірки. Це виключає темпоральний витік, коли майже ідентичні суміжні кадри, зняті з інтервалом у частки секунди, могли б опинитися одночасно в різних вибірках, штучно завищуючи метрики детектора та трекера.

Другим принципом є багатовимірна стратифікація. Оскільки випадковий розподіл не здатний збалансувати вибірки за умов сильного дисбалансу класів, для вирівнювання розподілу ознак було застосовано комплексний підхід. Він дозволяє одночасно враховувати геометрію цілі, тип сенсора камери, складність тла та часові характеристики відеоепізодів під час формування вибірок.

Для врахування різної тривалості відео застосовано метод квантильного аналізу. Для кожного відеоепізоду обчислюється точна кількість кадрів, після чого за методом рівночастотного групування вони розподіляються на короткі, середні та тривалі. На основі поєднання класу

метаданих та отриманого інтервалу тривалості формується інтегральний вектор стратифікації.

Застосування алгоритму стратифікованого вибору до деталізованих комбінованих класів супроводжується ризиком математичного виродження розділення, якщо певна комбінація містить недостатню кількість зразків. Для вирішення цієї проблеми було реалізовано двохетапний адаптивний алгоритм спрощення.

Алгоритм аналізує частоту появи кожного деталізованого класу. Якщо семантичний клас містить менше 6 відео в усьому наборі даних, такі епізоди примусово і в повному обсязі спрямовуються до навчальної вибірки. Це дозволяє зберегти рідкісні критично важливі ознаки для навчання моделі та запобігає викривленню оцінки на тестовому спліті, де пропуски чи хибні спрацьовування на одиничних кадрах викликали б штучний колапс метрик.

За допомогою описаного методу багатовимірної стратифікації було розподілено 594 унікальні відеоепізоди. Результати збалансованого розбиття за трьома доменами метаданих наочно представлено на рисунку 3.3.

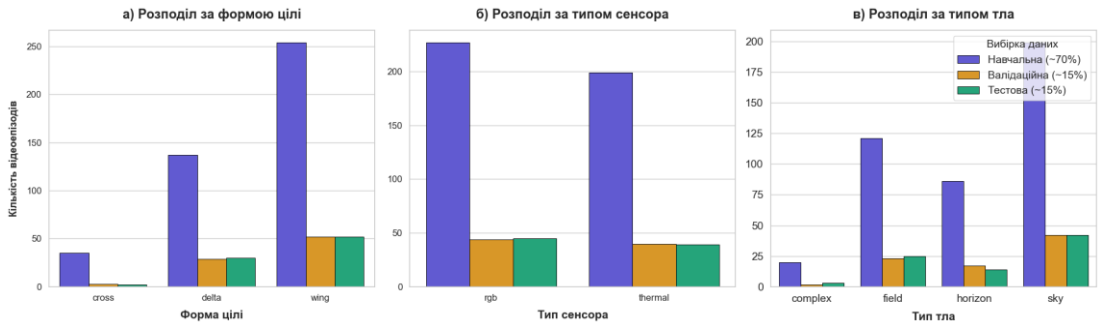


Рисунок 3.3 – Розподіл унікальних відеоепізодів у наборі даних за категоріями метаданих: а) за формою цілі; б) за типом сенсора; в) за типом тла

Загальний обсяг сформованого набору даних становить 594 відеоепізоди, кількісний склад яких розподілено таким чином:

– навчальна вибірка – 420 відеоепізодів;

– валідаційна вибірка – 87 відеоепізодів;

– тестова вибірка – 87 відеоепізодів.

Аналіз результатів розподілу підтвердив високу репрезентативність підходу. Для всіх масових категорій метаданих похибка відхилення від цільової пропорції 70:15:15 не перевищила 0.5%. Зокрема, для найбільш представленого класу, літакове крило на тлі неба в оптичному спектрі, розподіл склав: 69.5% у тренувальній вибірці, 15.3% у валідаційній та 15.3% у тестовій. Для тепловізійного аналогу цього класу пропорція дотримана на рівні: 69.9% у тренувальній вибірці, 14.5% у валідаційній та 15.7% у тестовій.

Усі унікальні та рідкісні класи, обсяг яких в усьому наборі даних становив менше 6 відео, були на 100% успішно перенесені в групу навчання. Це повністю захистило обчислювальний контур моделі від виникнення помилок ділення на нуль під час валідації та забезпечило збереження унікальних бойових ознак для етапу навчання детектора.

Сформовані вибірки були фізично розподілені по ізольованих каталогах. Для забезпечення відкритості та зручного відтворення експериментів підготовлений набір даних було розміщено у відкритому репозиторії Hugging Face, що спрощує імпорт та розгортання вибірок у хмарних обчислювальних середовищах.

3.8 Напівавтоматизоване анотування, верифікація та структурування даних

Формування репрезентативного набору даних для систем комп'ютерного зору вимагає створення точної та повної просторової розмітки для кожного кадру. Для оптимізації цього трудомісткого процесу в межах роботи було розроблено та застосовано триетапну схему напівавтоматизованого маркування, верифікації та структурування

накопичених кадрів. Перед безпосереднім початком побудови розмітки необхідно детально оцінити загальну трудомісткість цього завдання, визначити його технічні обмеження та обґрунтувати вибір гібридного підходу.

3.8.1 Оцінка складності та обґрунтування вибору гібридного підходу

Формування високоякісного набору даних для систем машинного зору є одним із найбільш трудомістких етапів розробки. Після ручного очищення та дедуплікації вихідних відеоепізодів загальний обсяг первинної бази даних склав 594 унікальні відеопослідовності сумарною тривалістю 38 хвилин. З урахуванням середньої частоти кадрів відеопотоку, загальний обсяг матеріалу для розмітки становить приблизно 69 402 кадри.

Повністю ручне маркування такої кількості зображень за допомогою нанесення обмежувальних рамок є нераціональним в умовах обмеженого часу проєктування. Навіть за консервативної оцінки у 5–10 секунд на ручне виділення одного об'єкта на кадрі, загальні часові витрати на повну розмітку набору перевищили б 100–190 годин безперервної праці.

Класичні напівавтоматичні інструменти розмітки, що інтегровані у більшість сучасних середовищ анотування, використовують лінійну екстраполяцію або інтерполяцію між ключовими кадрами. Проте у задачах повітряного перехоплення БПЛА такі підходи демонструють низьку ефективність через складну кінематику повітряного бою. Цілі рухаються з високою кутовою швидкістю, виконуючи різкі маневри нахилу та крену, що постійно змінює пропорції їхнього силуету, зближення перехоплювача з ціллю на зустрічних курсах зумовлює експоненціальне збільшення масштабу об'єкта в кадрі.

За таких умов лінійне перенесення рамки за замовчуванням вимагає постійного ручного коригування майже на кожному другому кадрі

відеопослідовності. Це нівелює часову перевагу екстраполяції та часто споживає більше ресурсів, ніж малювання рамки з нуля.

Для подолання цього бар'єра було прийнято рішення застосувати концепцію Human-in-the-Loop (людина в контурі керування), побудовану на двохетапній схемі. На першому етапі здійснюється автоматична генерація первинних псевдоанотацій за допомогою сучасної нейромережевої моделі трекінгу об'єктів на відео, а на другому – наступна ручна перевірка, коригування меж рамок та валідація отриманих даних у спеціалізованому середовищі розмітки. Такий підхід дозволив успішно поєднати швидкість нейромережевих алгоритмів із високою надійністю експертного контролю.

3.8.2 Генерація псевдоанотацій за допомогою моделі SAM 3

Для генерації первинних псевдоанотацій було застосовано мультимодальну нейромережу SAM 3. Вибір цієї моделі зумовлений тим, що попередні версії подібних алгоритмів вимагали попереднього ручного втручання для виділення об'єкта на першому кадрі шляхом нанесення точок або малювання рамки, щоб задати візуальний орієнтир для відстеження. SAM 3 усуває цей часомісткий етап, дозволяючи ініціалізувати процес автономно через текстовий опис цілі.

Модель SAM 3 здатна відстежувати об'єкти на відео на основі їхнього текстового опису. Завдяки тому, що архітектура мережі співвідносить семантичний зміст слів із відповідними візуальними ознаками на зображенні, вона розпізнає вказаний текстовий запит і самостійно локалізує відповідний силует у повітрі на початкових кадрах відео, після чого продовжує супроводжувати його упродовж усього відеоряду.

Через високу обчислювальну складність моделі SAM 3 процес генерації псевдоанотацій було перенесено у хмарне середовище Kaggle з використанням графічного прискорювача Nvidia Tesla T4. Для забезпечення стабільної обробки великих обсягів даних було розроблено наскрізний

програмний конвеєр з інтегрованим автономним механізмом синхронізації сесій. З метою запобігання втраті прогресу в разі обриву мережевого з'єднання або примусового перезапуску віртуальної машини у хмарному сховищі Hugging Face зберігається текстовий файл журналу обробки. Перед кожним запуском розроблений скрипт зчитує цей файл, визначає поточну чергу необроблених відео, а після завершення обробки чергової групи з N файлів оновлює та перезаписує лог у сховищі, що дозволяє безпечно продовжувати роботу з місця зупинки.

На основі згенерованих масок розроблений Python-скрипт розраховував крайні кутові координати обмежувальної рамки дрона ($x_{min}, y_{min}, x_{max}, y_{max}$). Для сумісності з інструментом CVAT ці координати програмно перераховувалися у формат MOT 1.1 (програмну реалізацію генератора псевдоанотацій та алгоритму конвертації у формат MOT 1.1 наведено у додатку В), де для кожного кадру записувався рядок із параметрами: frame_id, track_id, x_min, y_min, width, height, 1, 1, 1.

Усі згенеровані файли програмно групувалися за батчами та архівувалися у ZIP-пакети. Після цього сформовані архіви завантажувалися у хмарне сховище Hugging Face, що забезпечило їхнє надійне централізоване збереження та повну готовність до подальшого імпорту в CVAT.

3.8.3 Коригування анотацій у середовищі CVAT

Автономна генерація псевдоанотацій є лише проміжним етапом, оскільки отримані автоматично дані потребують обов'язкової експертної валідації та виправлення помилок. Для забезпечення зручного та швидкого процесу верифікації розмітки було спроєктовано та програмно реалізовано двонапрямний програмний міст між хмарним репозиторієм Hugging Face та локально розгорнутим сервером анотування CVAT.

Цей інструмент повністю автоматизував рутинні операції імпорту та експорту даних через командний інтерфейс cvat-cli та бібліотеку HTTP-запитів requests, виключивши потребу в ручному завантаженні та налаштуванні кожного окремого завдання.

Процес підготовки завдань для експертної розмітки відбувався послідовно за таким алгоритмом. Спочатку програмний міст звертався до репозиторію Hugging Face, зчитував структуру папок конкретного пакета даних та завантажував локально вихідне відео і відповідний архів первинної розмітки у форматі MOT 1.1. Після цього через утиліту cvat-cli на локальному сервері CVAT програмно створювалася нова задача (Task) з унікальним ідентифікатором (Task ID), назва якої автоматично пов'язувалася з назвою відео для уникнення хаосу в метаданих. На завершення на створену задачу програмно завантажувався вихідний відеофайл, а слідом – архів згенерованих моделлю SAM 3 псевдоанотацій.

Після завершення ручного виправлення всіх кадрів у межах завантаженого батчу, програмний міст виконував зворотну операцію. Спершу скрипт відправляв HTTP-запит до REST API CVAT, зчитував метадані задачі за її Task ID та визначав оригінальну назву відеоепізоду. Далі програмно запускався процес експорту виправлених, чистих анотацій у форматі MOT 1.1 у локальний ZIP-архів. На фінальному етапі отриманий архів із верифікованою розміткою програмно завантажувався у хмарне сховище Hugging Face у відповідну директорію вибірки та батчу. Для очищення анотацій перед завантаженням було реалізовано програмну перевірку унікальності ідентифікаторів супроводження (Track ID) в межах кожного епізоду. З огляду на те, що в полі зору камери протягом усього епізоду завжди перебуває лише один дрон, а модель SAM 3 іноді припускалася темпорального дрейфу і могла призначити одному й тому самому об'єкту різні ідентифікатори при повторному захопленні, скрипт примусово уніфікував усі траєкторії до єдиного індексу, що гарантувало наявність строго однієї цільової траєкторії в межах стандарту MOT 1.1.

Завдяки реалізації цього двонапрямного конвеєра вдалося організувати надійний, швидкий та захищений від втрати даних процес верифікації розмітки для всього масиву відео.

3.8.4 Екстракція кадрів та конвертація анотацій у формат YOLO

Перехід від неперервних відеопослідовностей, структурованих для задач темпорального трекінгу у форматі MOT 1.1, до ізольованих зображень, оптимізованих під одноетапні архітектури детекції сімейства YOLO, потребує системного перетворення координат та покадрової сегментації. Цей процес поєднує математичну нормалізацію просторових меж об'єктів та розділення кадрів на позитивні та негативні підмножини вибірки.

Вхідна розмітка у форматі MOT 1.1 визначає просторове положення об'єкта через абсолютні піксельні координати верхнього лівого кута прямокутник та його лінійні розміри – ширину і висоту.

Для навчання моделей YOLO ці параметри необхідно перерахувати у нормовані безрозмірні величини відносно центру обмежувальної рамки та її відносних лінійних пропорцій. Математичне перетворення для кожного кадру з урахуванням поточної ширини W та висоти H зображення здійснюється за такою системою рівнянь:

$$x_n = \frac{x_{left} + \frac{w}{2}}{W}, \tag{3.3}$$

$$y_n = \frac{y_{top} + \frac{h}{2}}{H}, \tag{3.4}$$

$$w_n = \frac{w}{W}, \tag{3.5}$$

$$h_n = \frac{h}{H}, \tag{3.6}$$

де x_n, y_n – нормовані координати центру обмежувальної рамки;

x_{left}, y_{top} – абсолютні піксельні координати верхнього лівого кута рамки;

w, h – абсолютна ширина та висота обмежувальної рамки в пікселях;

W, H – загальна ширина та висота зображення в пікселях;

w_n, h_n – нормована ширина та висота обмежувальної рамки відповідно.

Отримані нормовані координати записуються разом із числовим індексом класу цілі, де кожному безпілотному літальному апарату примусово присвоюється значення єдиного цільового класу.

Для формування первинного неструктурованого набору даних виконується покадровий аналіз кожної відеопослідовності. Для кожного кадру перевіряється наявність відповідних просторових анотацій у верифікованому наборі даних. Якщо на поточному кадрі зафіксовано присутність цілі, зображення експортується до підмножини позитивних зразків, а перераховані нормовані координати записуються у текстовий файл із розширенням .txt. Оскільки формат анотування YOLO вимагає наявності двох файлів для кожного позитивного кадру, зображення .jpg та файлу міток .txt, загальна кількість позитивних кадрів після екстракції дорівнює половині сумарного обсягу згенерованих файлів у відповідних директоріях.

У випадку повної відсутності анотації для кадру, коли безпілотний апарат ще не з'явився у полі зору камери або вже перетнув її межі, зображення класифікується як природний негативний зразок і зберігається у підмножині негативних зразків без файлу міток. Кількісні параметри отриманого масиву даних безпосередньо після завершення етапу покадрової екстракції наведено в таблиці 3.1.

Таблиця 3.1 – Кількісні параметри вибірок безпосередньо після покадрової екстракції

Назва вибірки	Позитивні кадри	Негативні кадри	Загальний обсяг кадрів
Навчальна	22 512	1533	24 045
Валідаційна	3953	142	4095
Тестова	4239	294	4533
Усього	30 704	1969	32 673

3.8.5 Методологія інтеграції та синтетичного моделювання негативних зразків

Для побудови надійної системи повітряного перехоплення критично важливо навчити нейромережу ігнорувати елементи OSD-телеметрії та статичні логотипи військових підрозділів. Якщо навчати модель виключно на кадрах із цілями, виникає ефект навчання легкими шляхами – мережа почне асоціювати наявність графічного інтерфейсу на екрані з присутністю БПЛА. Для розриву цього помилкового логічного зв'язку обов'язковим є інтеграція негативних зразків - кадрів, що містять завади та інтерфейс, але повністю позбавлені цілей.

Первинний масив природних негативних кадрів, автоматично отриманий після покадрової екстракції позитивних відеопослідовностей (таблиця 3.1), становив 1969 одиниць. З огляду на те, що кадри порожнього тла мають вкрай високу перцептивну схожість, під час подальшої математичної дедуплікації значна їхня частина неминуче буде відсіяна алгоритмами стиснення як надлишкові повтори, що може призвести до дефіциту завадостійких зразків у навчальному наборі.

Для запобігання цьому виникла потреба штучного розширення негативної вибірки. Головною перешкодою під час збору таких матеріалів є

дефіцит чистих фонових записів у військових OSINT-джерелах, оскільки підрозділи публікують у відкритий доступ переважно успішні епізоди ураження цілей. Крім того, залучені кадри повинні строго відповідати динамічній перспективі «повітря-повітря», що унеможливлює використання стандартних наборів даних знімання неба з землі.

Шляхом суцільного пошуку у відкритих джерелах було відібрано 7 унікальних фонових відеозаписів, які повністю відповідали кінематиці польоту та кутам огляду позитивної вибірки. Щоб залучені порожні фонові відео виконували свою регуляризаційну роль, на них було програмно накладено статичні копії найпоширеніших вотермарок.

Покадрова екстракція з цих семи синтезованих відеопотоків дозволила сформувати первинний набір негативних зразків загальним обсягом 8 119 кадрів, з яких 5 938 кадрів було спрямовано у тренувальну вибірку, 1 047 кадрів – у валідаційну, а 1 134 кадри – у тестову.

Цей масив створювався як первинний резервуар даних, який на наступних етапах обробки пройде через алгоритми дедуплікації для формування компактного, неперенасиченого та завадостійкого набору фонових зображень.

3.9 Триетапний алгоритм дедуплікації даних на рівні розмічених кадрів

Для усунення часової та візуальної надлишковості, а також запобігання деградації архітектури нейромережі, було розроблено триетапну методологію дедуплікації кадрів навчальної вибірки. Цей підхід дозволяє оптимізувати обсяг даних перед безпосереднім початком навчання моделі YOLO. Для розуміння архітектури розробленого конвеєра спершу необхідно проаналізувати обмеження, що притаманні традиційним підходам до дедуплікації.

3.9.1 Обмеження класичних методів дедуплікації

Для формування репрезентативного набору даних необхідно усунути часову та візуальну надлишковість у вихідних матеріалах. Сусідні кадри відео, записаного з високою частотою, містять велику кількість повторюваної інформації. Проте застосування стандартних методів фільтрації у досліджуваній задачі супроводжується низкою обмежень.

Першим суттєвим обмеженням є неефективність рівномірного темпорального проріджування відеопотоку через відбір кожного N-го кадру. Військові OSINT-відеоматеріали характеризуються нерівномірною динамікою. Деякі епізоди на стадії виявлення або термінального зближення штучно уповільнені для деталізації моменту ураження, що генерує десятки майже ідентичних кадрів, тоді як інші відео є штучно прискореними або відображають природно швидкі епізоди зближення з високою кутовою швидкістю маневрування цілі. Рівномірний відбір за часовим кроком неминуче призведе до накопичення дублікатів у повільних сценах і водночас до критичної втрати унікальних геометричних фаз у швидких високодинамічних епізодах.

Іншим обмежувальним фактором виступає недостатня надійність класичних алгоритмів порівняння зображень, таких як перцептивне хешування або індекси структурної подібності. Ці методи є чутливими до локальних змін яскравості, колірного контрасту та цифрових завад. В умовах аналогового снігового шуму, компресійних макроблоків та постійного коливання сонячного освітлення згадані алгоритми класифікують абсолютно статичні кадри з дроном як унікальні, що унеможливлює проведення якісної дедуплікації.

Також не дозволяє досягти стабільного результату класичний оптичний потік. Математично оптичний потік описує поле векторів переміщення пікселів (u, v) між послідовними кадрами, задовольняючи рівнянню фотометричної сталості:

$$I(x, y, t) = I(x + u, y + v, t + \Delta t), \quad (3.7)$$

де $I(x, y, t)$ – яскравість пікселя з координатами x та y в момент часу t;

u, v – компоненти вектора зміщення пікселя за координатами x та y відповідно;

Δt – часовий інтервал між двома послідовними кадрами.

У задачах дедуплікації кадр зберігається лише тоді, коли інтегральна величина цього поля відносно попереднього кроку перевищує заданий поріг τ. Проте такий абсолютний поріг не є інваріантним до кінематики зближення цілей. Для епізодів із повільним наближенням медіанне значення потоку є суттєво нижчим, ніж для швидких, навіть за ідентичних просторових фаз перехоплення. Це призводить до системного недобору кадрів у повільних відео та надлишкового накопичення дублікатів у швидких. Крім того, існує розрив між фотометричною та семантичною відмінністю кадрів. Значний потік може генеруватися завадами, наприклад рухом хмар або зміною освітлення, за повної семантичної ідентичності сцени, тоді як важливе для навчання детектора наближення дрона на великій відстані створює мінімальне зміщення пікселів.

Окремою проблемою є непридатність стандартного використання глибоких семантичних ембедингів усього кадру на початкових етапах виявлення БПЛА, коли він розташований на великій відстані від камери і займає малу площу зображення. Згорткові шари кодувальника внаслідок послідовного зменшення просторової роздільної здатності повністю пригнічують слабкі локальні ознаки такої мікроцілі. Через це результуючий вектор несе семантичну інформацію виключно про підстильну поверхню кадру, ігноруючи дрон.

Для подолання цих обмежень було розроблено триетапний конвеєр дедуплікації на рівні розмічених кадрів.

3.9.2 Просторова дедуплікація за метрикою IoU

Першим кроком очищення є просторова дедуплікація суміжних кадрів у межах кожної окремої відеопослідовності. Для збереження часової зв'язності всі кадри конкретного епізоду сортуються в хронологічному порядку. Перший кадр приймається за якірний, а його обмежувальна рамка виступає як опорна. Для кожного наступного кадру розраховується показник перетину Intersection over Union (IoU) між його поточною рамкою та опорною рамкою якірного кадру (алгоритм просторової дедуплікації зображень та обчислення метрики перетину наведено у додатку Г). Вибір порогового значення $\mathrm{IoU} = 0{,}95$ обумовлений необхідністю відсікання виключно статичних дублікатів з урахуванням апаратних шумів стиснення відеопотоку. Експериментальне порівняння різних варіантів показало, що за нижчого значення, наприклад $\mathrm{IoU} = 0.90$, алгоритм починає помилково видаляти кадри, де ціль здійснює мікроскопічні просторові зміщення або змінює свій кутовий ракурс, що збіднює варіативність геометричних ознак БПЛА під час навчання. Натомість за вищого значення порогу ($\mathrm{IoU} = 0.98$), який є занадто близьким до абсолютної ідентичності, алгоритм стає нечутливим до статичних дублікатів через наявність високочастотного шуму кодека та завад зв'язку, які створюють штучні флуктуації координат обмежувальної рамки навіть на повністю нерухомих кадрах. Таким чином, поріг 0.95 є оптимальним компромісом, що надійно відфільтровує апаратний шум камери і зберігає реальну кінематику руху цілі. Якщо обчислений показник IoU перевищує цей встановлений поріг, кадр маркується як дублікат і програмно переміщується у смітник. Якщо ж $\mathrm{IoU} \leq 0.95$, поточний кадр стає новим якірним, а його обмежувальна рамка приймається як опорна для наступних порівнянь.

Цей метод дозволив ефективно усунути статичні ділянки уповільнених відеопослідовностей. За результатами виконання просторової дедуплікації на першому етапі у тренувальній вибірці було відсіяно 9796

із 45024 файлів, у валідаційній – 1872 із 7906 файлів, а в тестовій вибірці – 1731 із 8478 файлів.

Оскільки анотація у форматі YOLO передбачає два файли для кожного позитивного кадру, кількість відфільтрованих зображень дорівнює половині від обсягу вилучених файлів.

Аналогічний процес семантичної дедуплікації на другому етапі було проведено для негативних зразків тренувальної вибірки. Оскільки ці кадри повністю позбавлені цілей, обчислення двокомпонентних ембедингів не потребувало виділення локальних фрагментів та розрахунку просторових відступів. Для оптимізації обчислювальних витрат і заощадження пам'яті вихідні комбіновані вектори були програмно зрізані до перших 768 вимірів, що описують виключно структуру та текстуру фону.

Через відсутність рухомої цілі послідовні кадри порожнього неба чи однорідної земної поверхні характеризуються значно вищою перцептивною схожістю, ніж позитивні кадри. Для ефективної фільтрації таких монотонних фонових послідовностей поріг косинусної відстані для негативів було встановлено на експериментально підібраному рівні, що відповідав вищому ступеню схожості. Завдяки цьому з початкових 5938 первинно відібраних негативних кадрів тренувальної вибірки алгоритм k-Center Greedy успішно відсіяв 5128 дубльованих фонових зображень. На виході було отримано 810 найбільш унікальних та геометрично відмінних негативних зразків, які й утворили фінальну завадостійку вибірку для навчання моделі.

3.9.3 Семантична дедуплікація на основі зважених ембедингів

Для семантичного очищення кадрів у межах кожної послідовності алгоритмом k-Center Greedy застосовано метод двокомпонентного кодування, що запобігає згасанню ознак дрібних об'єктів у згорткових шарах. Розроблений підхід об'єднує два типи представлень: глобальний

контекст навколишнього середовища та локальну геометрію самого об'єкта. З цією метою для кожного кадру за допомогою нейромережі DINOv2 згенеровано два окремі L2-нормовані ембедінги: представлення всього кадру (emb_{full}), що описує тло кадру, та представлення вирізаного фрагмента цілі (emb_{crop}). Локальний фрагмент цілі вирізається із додатковим просторовим відступом для збереження контексту про контраст силуету БПЛА відносно тла та зміну його масштабу. Величина цього відступу динамічно адаптується під площу цілі: для мікроцілей вона збільшується до 4.0, щоб надати згортковим шарам достатньо пікселів тла для формування чіткого локального градієнта на межах об'єкта, тоді як для великих зблизька об'єктів відступ зменшується до 1.2, запобігаючи домінуванню текстури тла над детальними ознаками самого БПЛА.

Замість фіксованої кількості кадрів K, що викривила б репрезентативність динамічних сцен, для алгоритму k-Center Greedy впроваджено поріг косинусної відстані θ_1. Алгоритм покроково обирає найвіддаленіші кадри-центри, доки відстань від будь-якого залишкового кадру до найближчого обраного центру не стане меншою за θ_1, що означає, що весь відеоряд покритий обраними центрами з радіусом схожості θ_1.

Для пошуку оптимального порогу θ_1 було проведено аналіз косинусних відстаней усередині 50 випадкових відеопослідовностей (рисунок 3.4, а). Середнє значення відстаней між суміжними кадрами в цих епізодах склало 0.2827 за мінімуму 0.0011. Математично обчислена через другу похідну точка згину кумулятивної кривої збереження кадрів склала 0.042 (рисунок 3.4, б). Однак значення 0.042 є математично надто консервативним і залишає високу темпоральну надлишковість, оскільки за такої умови у вибірці зберігається понад 32% від загального обсягу кадрів.

Для точнішого калібрування було проаналізовано перцентилі розподілу відстаней між суміжними кадрами відео, де десятий перцентиль становить $p_{10} = 0.072$, а двадцять п'ятий – $p_{25} = 0.132$. З метою визначення оптимального робочого порогу було проведено експертний візуальний аудит відібраних кадрів для п'яти випадкових відеопослідовностей при трьох контрольних значеннях: $\theta_1 = 0.05$ (консервативний), $\theta_1 = 0.10$ (оптимальний) та $\theta_1 = 0.15$ (агресивний).

Результати аудиту показали, що при консервативному порогу $\theta_1 = 0.05$ у вибірці залишається значна кількість візуально ідентичних кадрів. За агресивного порога $\theta_1 = 0.15$ відбувається деградація вибірки через втрату важливих динамічних фаз маневрування дрона. Значення порогу $\theta_1 = 0.10$, яке перебуває в інтервалі між перцентилями p_{10} та p_{25}, забезпечило найкращий результат: воно повністю усунуло монотонність тла кадру, але зберегло всі унікальні просторові стани об'єкта. За цього порогу у вибірці залишається близько 8% найінформативніших кадрів, але повністю зберігаються всі унікальні фази маневрів, змін ракурсу та масштабу цілі.

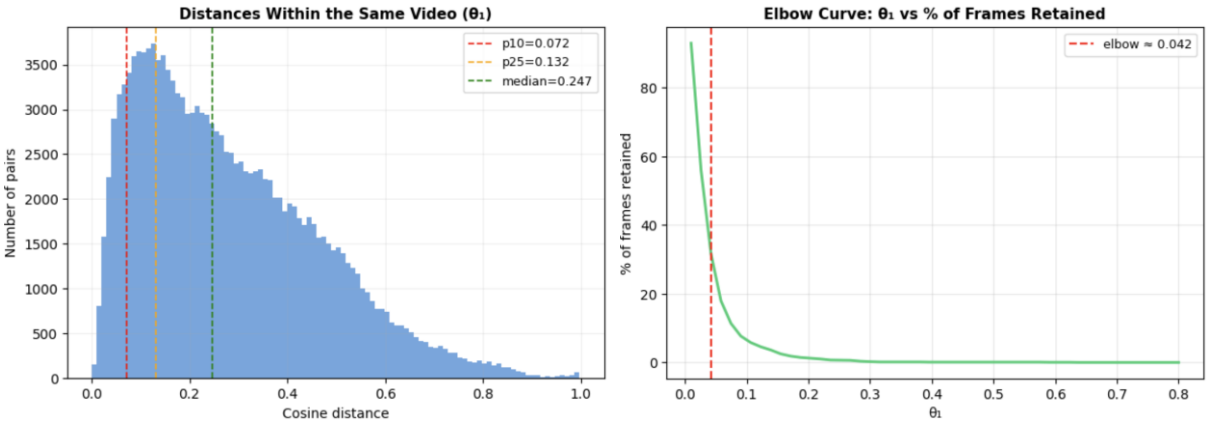


Рисунок 3.4 – Результати статистичного аналізу косинусних відстаней у межах одного відео: а) гістограма розподілу парних відстаней; б) крива залежності відсотка збережених кадрів від порогу

На другому етапі дедуплікації позитивів тренувальної вибірки було відсіяно 27 000 кадрів з 35 228, що залишило 8 228 унікальних кадрів.

3.9.4 Глобальна семантична дедуплікація на основі зважених ембедингів

Третім етапом розробленого конвеєра мала бути глобальна дедуплікація на рівні всього набору даних за порогом θ_2 для вилучення візуально подібних кадрів між різними відеофайлами. Оскільки на попередніх етапах було видалено значну кількість надлишкових кадрів і візуально вони демонстрували високу відмінність, було вирішено додатково перевірити потребу в проведенні глобальної дедуплікації. Для цього було здійснено статистичний аналіз косинусних відстаней між випадковими парами кадрів із різних відео після завершення другого етапу, результати якого спростували практичну необхідність подальшої фільтрації (рисунок 3.5).

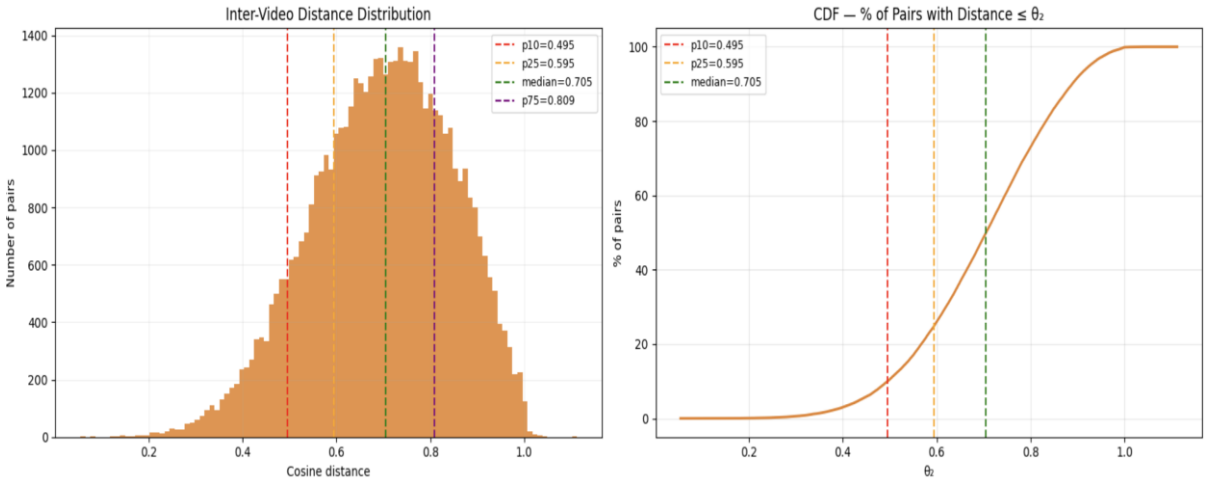


Рисунок 3.5 – Результати статистичного аналізу косинусних відстаней між різними відеофайлами після другого етапу дедуплікації: а) гістограма розподілу міжавторських відстаней θ_2; б) кумулятивна функція розподілу

Згідно з отриманими даними, середнє значення відстаней між різними відео становить 0.6969 (рисунок 3.5, а). На гістограмі розподілу цих відстаней чітко видно, що лівий хвіст починає зростати лише в діапазоні $0.35 - 0.40$, а до цієї межі кількість випадкових збігів є нульовою. Наявність косинусних відстаней між різними відео, що перевищують 0.35 для понад 99% випадкових пар (рисунок 3.5, б), математично доводить, що кадри з різних відеопослідовностей є унікальними.

Вони представляють принципово різні краєвиди підстильної поверхні, кути зйомки та погодні умови. Проведення третього етапу глобальної дедуплікації було відхилено як обчислювально надлишкове, оскільки воно вимагало б значних ресурсів на розрахунок повнозв'язної матриці схожості для всього набору даних без будь-якої практичної користі для якості навчання детектора.

Для забезпечення об'єктивності та реалістичності оцінювання розробленої системи тестувальна та валідаційна вибірки не піддавалися дедуплікації за алгоритмом k-Center Greedy, а натомість були сформовані за двокомпонентною схемою. Для верифікації та тестування безпосередньо моделі детектора було застосовано лише просторову фільтрацію першого етапу та рівномірне часове проріджування кадрів із їхньою подальшою конвертацією у формат YOLO.

Водночас для порівняльного аналізу та оцінювання стійкості алгоритмів супроводження за інтегральними метриками MOTA, IDF1 та HOTA у цих вибірках було повністю збережено оригінальні неперервні відеопослідовності разом із файлами траєкторій у форматі MOT 1.1. Така подвійна структура дозволила з одного боку уникнути перенасичення валідаційних метрик детектора дублікатами, а з іншого – провести достовірне тестування всієї системи детекції та супроводження БПЛА на реальних безперервних траєкторіях польоту.

3.10 Статистичний аналіз та верифікація дедуплікованого набору даних

Після завершення всіх етапів фільтрації, сегментації, анотування та дедуплікації сформований набір даних потребує ретельного всебічного аналізу та верифікації. Це необхідно для підтвердження його репрезентативності, збалансованості та відсутності прихованих аномалій перед початком практичного навчання нейромережевих моделей. У цьому підрозділі наведено результати статистичного оцінювання кількісних, семантичних та геометричних характеристик отриманого дедуплікованого датасету.

3.10.1 Кількісний аналіз та розподіл даних за вибірками

Для проведення об'єктивного навчання нейромережевої моделі детекції та подальшого порівняльного аналізу алгоритмів темпорального супроводження було сформовано кінцевий дедуплікований набір даних, загальний обсяг якого склав строго 13 200 кадри. З них 12 000 кадрів є позитивними зразками, а 1 200 кадрів – негативними. Кількісний розподіл кадрів між ізольованими вибірками наведено в таблиці 3.2.

Таблиця 3.2 – Кількісні параметри дедуплікованого набору даних

Назва вибірки	Позитивні кадри	Негативні кадри	Загальний обсяг кадрів
Тренувальна	8 177	810	8 987
Валідаційна	1 542	147	1 689
Тестувальна	2 281	243	2 524
Усього	12 000	1 200	13 200

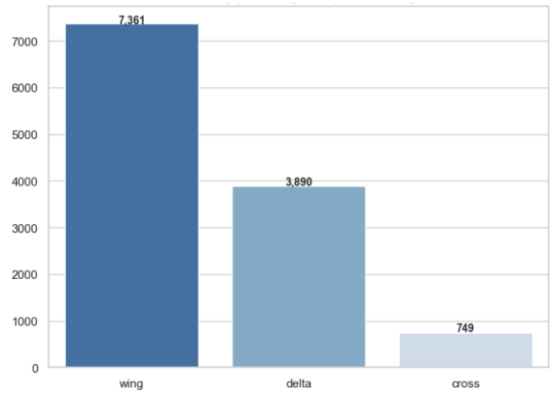
Проведений детальний статистичний аудит стратифікованого розподілу даних підтверджує, що сформована навчальна вибірка складає 68.1% від загального обсягу, валідаційна – 12.8%, а контрольна

тестова складає 19.1%, що з урахуванням загальної тривалості відео та складного поєднання унікальних метаданих є максимально наближеним до цільової пропорції 70:15:15. Поряд з цим, у межах кожної окремої вибірки стабільно утримується строгий баланс негативних кадрів на рівні 9.1% для тренувальної, 8.7% для валідаційної та 9.6% для тестувальної.

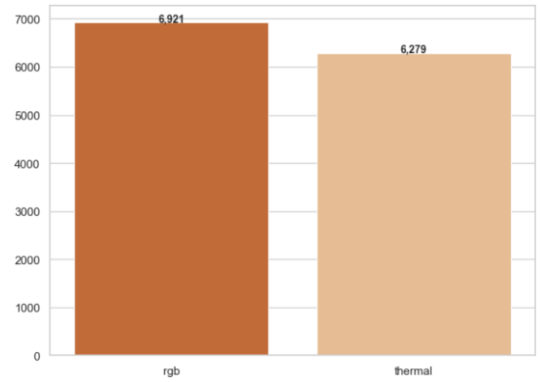
3.10.2 Семантичний розподіл метаданих дедуплікованих кадрів

Аналіз розподілу окремих категорій метаданих на рівні дедуплікованих кадрів (рисунок 3.6) дозволяє оцінити репрезентативність сформованої вибірки перед початком експериментальних досліджень. За критерієм геометричної форми цілі в позитивній вибірці зафіксовано 7 361 кадр із класичним літаковим крилом, 3 890 кадрів із дельтоподібним крилом та 749 кадрів із хрестоподібним силуетом. Такий дисбаланс категорій не є дефектом збору даних, а точно відображає статистичну реальність бойового повітряного простору, де літакоподібні розвідники та ударні БПЛА типу Shahed застосовуються масово, тоді як хрестоподібні баражуючі боєприпаси класу Lancet є значно рідкіснішими цілями.

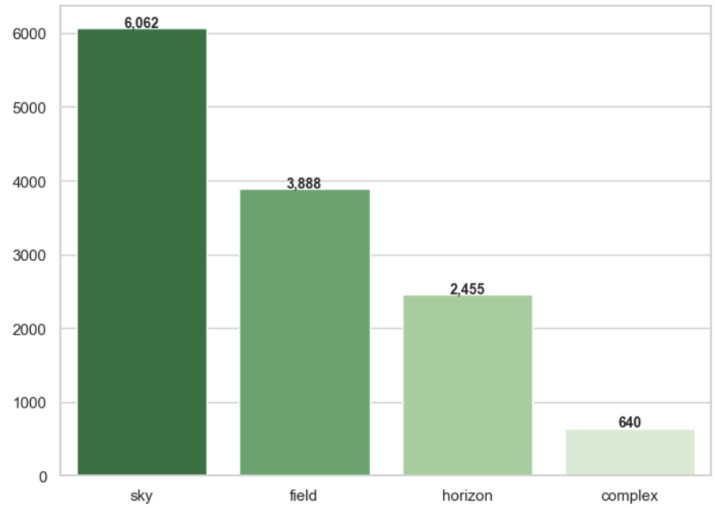
Проаналізований статистичний розподіл за видами використаних сенсорів у сформованому наборі даних є близьким до рівномірного: 6 921 кадр отримано з денних кольорових оптичних камер видимого діапазону, що становить 52,4% від загального обсягу, тоді як решта 6 279 кадрів, або 47,6%, припадає на інфрачервоні тепловізійні сенсори середньо- чи довгохвильового діапазону. Такий паритет гарантує, що навчена модель матиме високу завадостійкість як під час денного супроводу БПЛА з використанням хроматичних ознак, так і під час нічних перехоплень у тепловізійному спектрі, де класичні колірні канали повністю відсутні.



а) за формою цілі



б) за типом камери



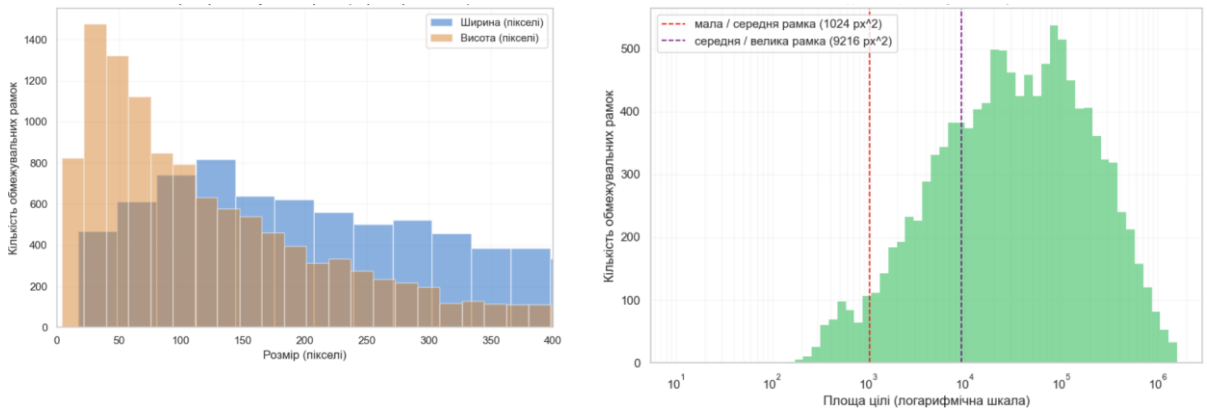
в) за типом фону

Рисунок 3.6 – Розподіл семантичних метаданих на рівні окремих дедуплікованих кадрів набору даних: а) за формою цілі; б) за типом камери; в) за типом фону

За класифікацією типів підстильної поверхні переважають зображення цілей на тлі відкритого неба та хмарних утворень, сумарна кількість яких становить 6 062 кадри. Водночас значна частина вибірки містить складніші кадри з положенням БПЛА під лінією горизонту на тлі полів у кількості 3 888 одиниць, безпосередньо на лінії динамічного переходу горизонту в обсязі 2 455 одиниць, а також на складному мозаїчному ландшафтному тлі антропогенного характеру, що поєднує лісові масиви, міську забудову, дорожні розв'язки та водні простори.

3.10.3 Геометричний аналіз обмежувальних рамок цілей

Проведений детальний статистичний аналіз ключових геометричних параметрів для масиву з 12 000 обмежувальних рамок (bounding boxes) цілей виявив такі характерні закономірності. Зокрема, середня ширина обмежувальної рамки становить 472,81 ± 413,07 пікселя, тоді як середня висота дорівнює 150,09 ± 144,38 пікселя (порівняльний розподіл лінійних розмірів наочно наведено на рисунку 3.7, а). Середнє інтегральне співвідношення ширини рамки до її висоти дорівнює 3,861, що безпосередньо вказує на характерну витягнуту по горизонталі форму більшості спостережуваних цілей.



а) б)

Рисунок 3.7 – Результати статистичного аналізу геометричних параметрів обмежувальних рамок цілей: а) порівняльний розподіл ширини та висоти рамок; б) розподіл площі рамок відносно порогових меж категорій розмірів COCO

Високі чисельні значення стандартного відхилення для показників ширини та висоти підтверджують присутність широкого динамічного масштабного діапазону цілей у сформованій вибірці. Він повноцінно охоплює діапазон від віддалених мікрооб'єктів після відсікання порогу

розміром 16 на 16 пікселів (рисунок 3.7, б) до великих деталізованих силуетів зблизька, що перекривають понад 50% загальної площі кадру на термінальній стадії перехоплення, що має гарантовано забезпечити майбутній глибокій моделі детекції необхідну масштабну інваріантність під час детекції.

3.10.4 Просторова верифікація центрів цілей

Побудована двовимірна теплова карта просторового розподілу центрів обмежувальних рамок виявила щільну вертикальну лінію концентрації цілей уздовж центральної горизонтальної осі кадру при максимальному накопиченні в його геометричному центрі (рисунок 3.8).

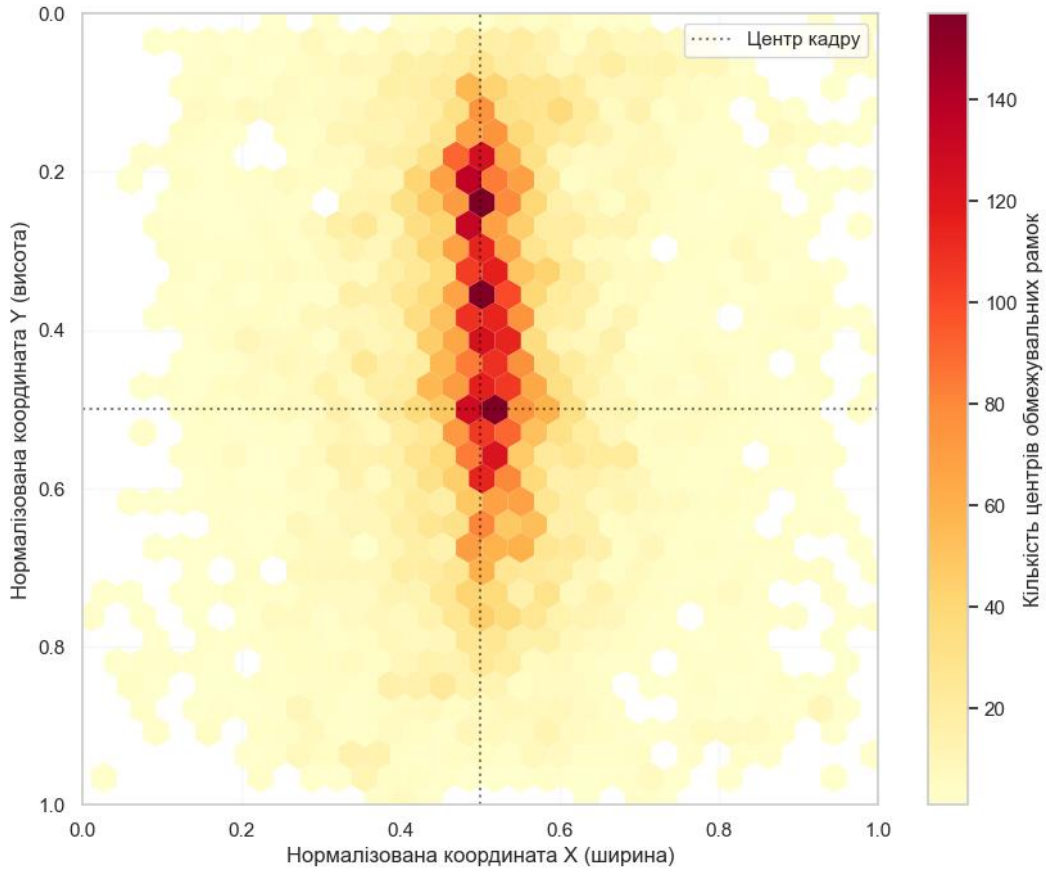


Рисунок 3.8 – Двовимірна теплова карта просторового розподілу центрів обмежувальних рамок БПЛА на нормованій площині кадру

З погляду машинного навчання, такий розподіл створює ризик виникнення стійкого просторового зсуву даних. Нейромережа може сформувати хибну залежність і шукати об'єкти виключно в центрі кадру, ігноруючи цілі на периферії на початкових етапах атаки, до моменту їхнього захоплення оператором. Цей аналіз доводить критичну необхідність майбутнього застосування під час навчання моделі просторових аугментацій зсуву зображення для примусового переміщення рамок цілей до країв кадру, що дозволить зробити модель стійкою до геометричного положення об'єкта.

3.10.5 Математичний аналіз локального контрасту цілі відносно тла

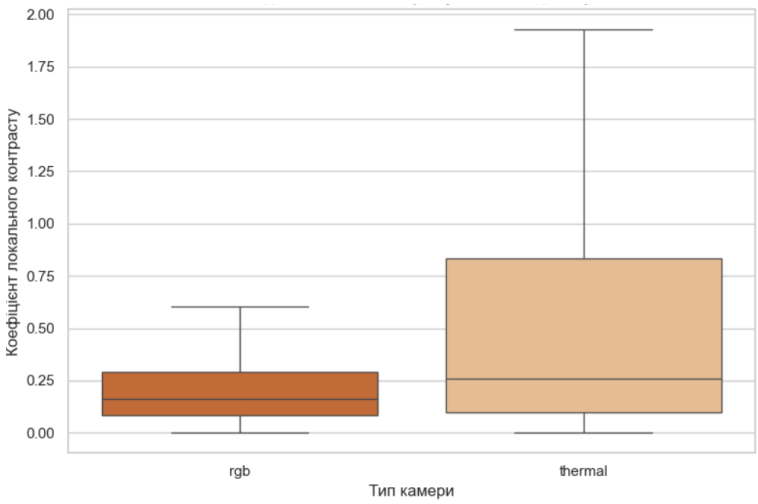
Проведений аналіз локального Веберового контрасту цілей відносно їхнього безпосереднього периферійного тла виявив суттєву різницю між двома спектральними діапазонами (рисунок 3.9, а). Вибір цієї моделі обґрунтований тим, що на відміну від контрасту Мікельсона, призначеного для періодичних або рівновеликих текстур, Веберовий контраст є стандартом для оцінювання видимості невеликих локальних об'єктів на великому однорідному тлі. Локальний контраст C обчислювався як відношення різниці яскравості цілі та тла до яскравості тла:

$$C = \frac{|\, I_{\text{target}} - I_{\text{bg}} \,|}{\max\,(1.0, I_{\text{bg}})}, \tag{3.8}$$

де I_{target} – середня яскравість пікселів у межах обмежувальної рамки цілі в одноканальному просторі градацій сірого;

I_{bg} – середня яскравість пікселів умовного кільця навколо неї, отриманого програмним розширенням рамки на 50% із вирізанням силуету самого БПЛА. Функція максимуму в знаменнику запобігає діленню на нуль у випадку повністю чорного тла.

а)



б)

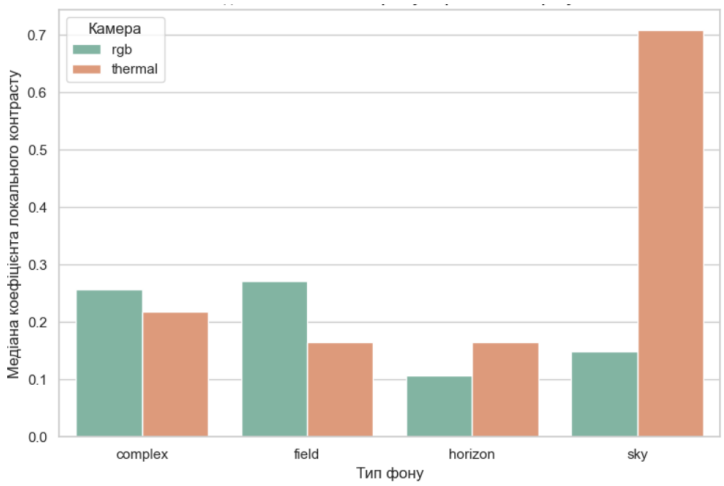


Рисунок 3.9 – Результати математичного аналізу локального контрасту цілі відносно навколишнього тла: а) порівняльний розподіл контрасту для оптичних та тепловізійних сенсорів; б) медіанний рівень контрасту цілей на різних типах підстильної поверхні

Для оптичного діапазону середній контраст становить $0{,}2284 \pm 0{,}2965$ при медіанному значенні $0{,}1628$, тоді як для тепловізійного діапазону середній показник дорівнює $1{,}9299 \pm 7{,}8700$ при медіанному значенні $0{,}2589$. Медіанний контраст в інфрачервоному спектрі є на 60% вищим, ніж у спектрі видимого світла, що підтверджує фізичну перевагу тепловізійної зйомки для виділення меж об'єктів. Помітною особливістю є значне середнє значення тепловізійного контрасту та його високе стандартне відхилення, що досягає $7{,}8700$. Це математично

доводить присутність у тепловізійній вибірці екстремальних відхилень, зумовлених як високою температурою робочих двигунів БПЛА на тлі холодного неба, так і зворотним контрастом – виявленням холодних планерів на тлі розігрітої сонячним випромінюванням земної поверхні. Такі чинники створюють високі локальні градієнти яскравості, що зміщує середнє значення вгору, хоча основна частина вибірки залишається зосередженою в межах медіанного значення.

Аналіз медіанного контрасту для різних типів підстильної поверхні показує, що у спектрі видимого світла контраст залишається стабільно низьким на всіх типах тла (рисунок 3.9, б). Натомість для тепловізійного діапазону спостерігається інша закономірність: на фоні неба показник контрасту досягає максимуму, але при наближенні до складного ландшафту чи поля він суттєво знижується, оскільки тепловий силует цілі починає частково зливатися з тепловим випромінюванням земної поверхні. Цей спад доводить необхідність використання алгоритмів локальної адаптивної корекції контрасту або гамма-корекції як обов'язкових інструментів попередньої обробки відеопотоку на борту перехоплювача для штучного посилення меж об'єктів у складних умовах видимості.

4 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ, ОПТИМІЗАЦІЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

Практична частина дослідження присвячена експериментальній перевірці розробленої методології підготовки даних та всебічній оцінці працездатності всього програмного конвеєра. У цьому розділі наведено результати навчання детектора YOLO26n та проведено діагностичний аналіз карт просторової уваги EigenCAM для оцінки стійкості моделі до ефекту навчання за спрощеними шляхами. Реалізовано низькорівневу оптимізацію детектора шляхом компіляції у формат TensorRT FP16 з детальним аналізом накладних витрат препроцесингу та системних обмежень квантування. Проведено порівняльний аналіз алгоритмів темпорального трекінгу в умовах шумних детекцій, а також розрахунку теоретичної наскрізної швидкодії всього конвеєра для підтвердження його апаратної життєздатності в режимі жорсткого реального часу на цільовій вбудованій платформі.

4.1 Конфігурація експериментальних досліджень та методологія пошуку навчання за спрощеними шляхами

На першому етапі експерименту було розгорнуто ізольований конвеєр навчання на базі хмарного середовища Kaggle з використанням графічного прискорювача Tesla T4 з об'ємом відеопам'яті 15 ГБ. Відповідно до обґрунтованого у підрозділі 2.1 вибору, як цільову архітектуру для експериментальних досліджень було задіяно одноетапний згортковий детектор YOLO26n, що характеризується мінімальною обчислювальною складністю та є оптимальним для подальшого розгортання на ресурсообмежених edge-пристроях.

Для забезпечення репрезентативності та чистоти експерименту було відключено всі стандартні геометричні, колірні та композиційні аугментації

під час навчання, оскільки вони б штучно зміщували або руйнували статичні водяні знаки та елементи OSD по краях кадру. Це частково замаскувало б ефект помилкового навчання за спрощеним шляхом (shortcut learning) і не дозволило б об'єктивно оцінити, наскільки сильно згорткова мережа схильна запам'ятовувати статичну графіку замість унікальної морфології цілі (програмна реалізація наведена у додатку Б).

Навчання моделей проводилося при фіксованій роздільній здатності вхідного тензора 1280x1280 пікселів, що є критично важливим для збереження дрібних просторових ознак малорозмірних цілей на великих відстанях. Весь тренувальний процес було обмежено 25 епохами з розміром пакета даних, що дорівнював 8, та порогом ранньої зупинки у 15 епох для запобігання перенавчанню. Оптимізацію параметрів мережі було покладено на алгоритм AdamW з адаптивним оцінюванням моментів та розділеним затуханням ваг, який був автоматично визначений середовищем як найбільш ефективний для цієї конфігурації. Початкова швидкість навчання становила $\mathrm{lr}_0 = 0.002$ при значенні імпульсу $\mathrm{momentum} = 0.9$.

Дослідження впливу структури даних на якість розпізнавання цілей базувалося на аналізі двох конфігурацій, де перша модель тренувалася виключно на дедуплікованих позитивних кадрах із подальшою валідацією на чистих позитивних зразках без негативних прикладів, тоді як друга модель навчалася на суміші позитивних кадрів та доданих негативних зразків із проведенням валідації на репрезентативному наборі, що містив обидві категорії даних.

Для забезпечення абсолютної порівнянності та репрезентативності результатів експериментів обидві моделі детекції було навчено за строго ідентичних умов на однакових апаратних ресурсах із використанням уніфікованих гіперпараметрів навчання.

Для проведення візуального аналізу було розроблено діагностичний скрипт (програмну реалізацію алгоритму візуалізації карт просторової уваги EigenCAM наведено у додатку И), який знаходить фінальний згортковий

шар просторового представлення ознак – шар SPPF у бекбоні YOLO26n – і за допомогою алгоритму EigenCAM генерує карти просторової уваги моделі. Вибір саме цього алгоритму є принциповим інженерним рішенням. На відміну від класичного методу Grad-CAM, який спирається на зворотне поширення градієнтів відносно оцінки конкретного класу, алгоритм EigenCAM повністю не залежить від градієнтів та прогнозів фінальної класифікаційної голови детектора.

Його математична логіка базується на сингулярному розкладі матриці активацій цільового згорткового шару та виділенні першої головної компоненти, яка репрезентує найбільш значущі просторові ознаки. Це робить EigenCAM оптимальним інструментом для об'єктивного виявлення навчання за спрощеними шляхами: навіть коли модель повністю пропускає дрон і видає нульовий результат детекції, алгоритм все одно показує, у які саме геометричні області кадру змістився фокус згорткових фільтрів бекбону. Згенеровані карти просторової уваги проекційним методом накладаються на випадкову вибірку кадрів, наочно демонструючи реальні орієнтири нейромережі під час роботи.

4.2 Аналіз карт активації EigenCAM та дослідження помилкових кореляцій

Як свідчать результати кількісного оцінювання, наведені у таблиці 4.1, інтеграція масиву негативних зразків до навчальної вибірки призвела до незначного зниження всіх базових метрик ефективності. Модель 1, яка навчалася виключно на позитивних прикладах, демонструє вищі показники за всіма метриками якості, ніж Модель 2, до якої було інтегровано негативні зразки. Точність вища на 1.35%, повнота – на 3.45%, а інтегральний показник mAP50 сягає 0.9178 порівняно з 0.8993.

Таблиця 4.1 – Показники ефективності нейромережевих моделей детекції YOLO26n

Конфігурація навчання	Точність	Повнота	mAP50	mAP50-95
Модель 1 (лише позитиви)	0.9078	0.8615	0.9178	0.6896
Модель 2 (позитиви та негативи)	0.8943	0.8270	0.8993	0.6727

Для оцінки стійкості детектора до доменного зсуву було проведено аналіз просторового розподілу ознак обох навчених конфігурацій. На рисунку 4.1 наведено отримані теплові карти активації EigenCAM, а в таблиці 4.2 представлено відповідні показники рівня впевненості детекції моделей.

Візуалізації Моделі №2 демонструють вищу інтенсивність активацій на водяних знаках та OSD-елементах порівняно з базовою Моделлю №1 (на кадрах «б», «г», «д»). При цьому хибні спрацювання на цих артефактах відсутні. Це пояснюється дією механізму активного пригнічення та специфікою EigenCAM, який відображає головну компоненту активацій усього тензора ознак (зони максимальної обчислювальної роботи згорткових фільтрів).

Таблиця 4.2 – Порівняльний огляд рівня довіри та виявлення

Назва кадру	Модель №1	Модель №2
а)	0.86	0.87
б)	0.55	0.90
в)	0.49	-
г)	0.69	0.9
д)	0.90	-

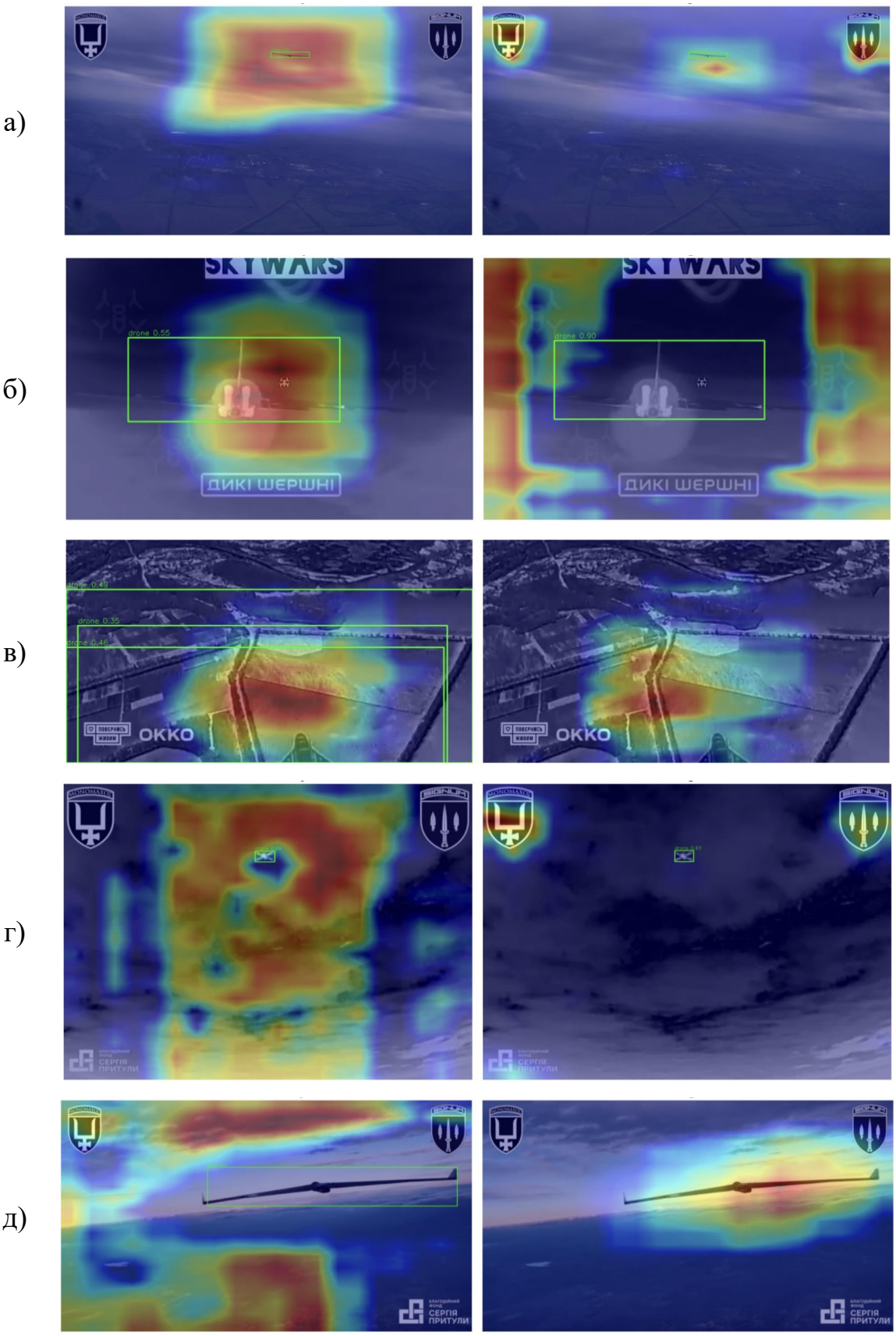


Рисунок 4.1 – Порівняння теплових карт активації EigenCAM: ліворуч – Модель №1, праворуч – Модель №2

Наявність у навчальній вибірці фонових кадрів із завадами змусила Модель №2 розпізнавати ці графічні елементи як нецільові. Висока активація в зонах тексту свідчить про успішну ідентифікацію завади з цілеспрямованим зниженням порогу об'єктності до нуля для відповідних просторових координат.

Натомість Модель №1, навчена без негативних зразків, демонструє розмиту просторову увагу. Нездатність дискримінувати артефакти спричиняє хибні спрацювання на складному тлі (кадр «в», хибні детекції наземних об'єктів та логотипа), тоді як Модель №2 повністю ігнорує ці завади (програмна реалізація наведена у додатку В).

Отже, інтеграція негативних зразків ускладнює задачу оптимізації, що обумовлює зниження метрики mAP50 з 0.9178 до 0.8993, проте формує механізм фільтрації артефактів.

4.3 Конфігурація процесу оптимізації та аналіз динаміки навчання моделі

На основі результатів попередніх експериментів та аналізу карт просторової уваги EigenCAM було розроблено конфігурацію навчання фінальної моделі детекції YOLO26n (програмну реалізацію конвеєра навчання та оцінювання якості моделі наведено у додатку Е). Головною метою цього етапу стало досягнення стійкого оптимуму узагальнюючої здатності нейромережі шляхом навчання на оригінальному наборі даних, який містив як позитивні кадри, так і інтегровані негативні фонові зображення.

Зважаючи на високу обчислювальну складність процесу обробки відеоданих високої роздільної здатності, тренування здійснювалося у хмарному середовищі із застосуванням графічного прискорювача Nvidia Tesla T4. Тренувальний процес було обмежено 50 епохами. Параметр

критерію ранньої зупинки було встановлено на рівні 15 епох, що запобігало передчасному перериванню тренування під час тимчасових плато збіжності.

Просторову роздільну здатність вхідного тензора було встановлено на рівні 1280×1280 пікселів, що вдвічі перевищує стандартний базовий розмір. Це компромісне інженерне рішення призводить до пропорційного зростання обчислювального навантаження під час автономної роботи, проте воно є необхідним для збереження високочастотних ознак малорозмірних цілей. Збільшення розміру вхідного зображення забезпечує значне розширення дистанції гарантованого виявлення дрона, надаючи системі автоматичного супроводження додатковий час для розрахунку вектора перехоплення.

Важливим кроком під час налаштування конвеєра даних стала повна відмова від застосування просторових та колірних аугментацій. Оскільки вхідні матеріали репрезентують нейтральний фізичний відеосигнал бортової камери, додаткове внесення штучного шуму або імітація радіозавад призвели б до небажаного доменного зсуву. Застосування алгоритмів композиційного змішування, таких як просторова мозаїка (Mosaic) або композиційне змішування (MixUp) було виключено через високу ймовірність критичного пошкодження або повного видалення силуетів мікроцілей під час випадкового обрізання кадрів.

У перспективі подальших досліджень доцільно провести окремий експериментальний аналіз спеціалізованих методів збагачення даних. Зокрема, перспективним вбачається застосування просторово-обмеженого маскування по периметру кадру для динамічного руйнування графічних завад, а також часткового маскування центру великих цілей для навчання мережі розпізнавати об'єкт за окремими елементами конструкції. Окремої уваги потребує специфіка просторового розподілу цілей: як було виявлено раніше на двовимірній тепловій карті (рисунок 3.8), у наборі даних наявна виражена вертикальна концентрація центрів обмежувальних рамок. Така позиційна упередженість вибірки математично обґрунтовує доцільність подальшого застосування алгоритмів горизонтального віддзеркалення,

випадкового масштабування та просторового зміщення для підвищення просторової інваріантності детектора.

Результати процесу навчання моделі впродовж 50 епох наведено на рисунку 4.2.

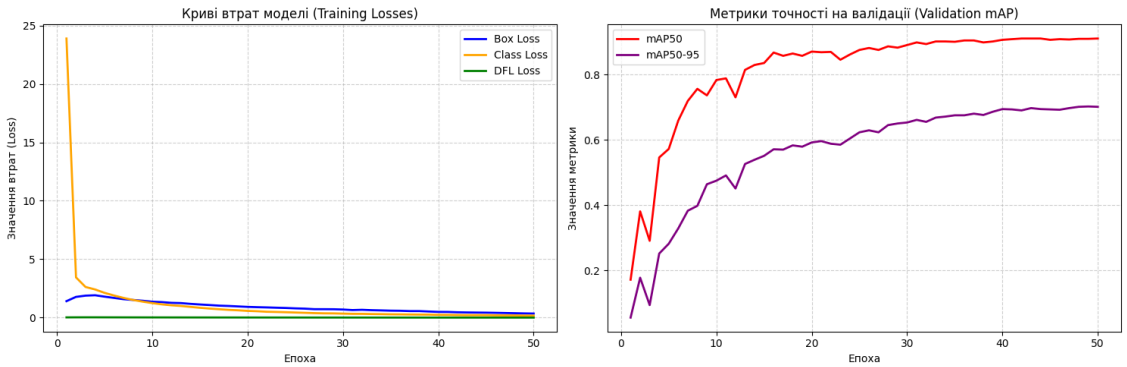


Рисунок 4.2 – Графіки процесу навчання фінальної моделі YOLO26n

Аналіз отриманих кривих функцій втрат (рисунок 4.2, ліворуч) свідчить про стабільність процесу збіжності моделі. Зокрема, втрати класифікації (Class Loss) зазнали стрімкого падіння вже після перших епох навчання, знизившись із початкового значення близько 24.0 до 3.5, після чого плавно наблизилися до показників, близьких до нуля, що вказує на адаптацію архітектури до монокласової задачі. Водночас втрати локалізації (Box Loss) продемонстрували незначне стартове зростання з 1.3 до 1.8 протягом перших п'яти епох, що є характерним процесом під час початкового прогріву параметрів під нестандартні витягнуті пропорції БПЛА, після чого цей показник плавно знизився до рівня близько 0.3. Що стосується втрат розподілу (DFL Loss), то вони мають вкрай низькі стабільні значення і візуально відображаються як горизонтальна лінія поблизу нуля впродовж усього процесу навчання, оскільки модуль Distribution Focal Loss не використовувався в архітектурі YOLO26n.

Динаміка метрик точності на валідаційній вибірці (рис. 4.2, праворуч) підтверджує правильність вибору тривалості навчання. Спостерігається

стрімке зростання показників mAP50 та mAP50-95 протягом перших 20 епох. Починаючи з 40-ї епохи, криві виходять на стабільне горизонтальне плато, mAP50 стабілізується на рівні понад 0.90. Це математично доводить, що модель досягла свого максимуму на наданих даних, не перейшовши у стан перенавчання, а обмеження у 50 епох було абсолютно раціональним.

Результати оцінювання ефективності розробленого детектора на ізольованій валідаційній та незалежній тестовій вибірках наведено в таблиці 4.3.

Таблиця 4.3 – Показники ефективності детектора на валідаційній та тестовій вибірках

Метрика	Валідаційна вибірка	Тестова вибірка
Точність	0,9279	0,8886
Повнота	0,8426	0,8144
mAP_{50}	0,9094	0,8668
mAP_{50-95}	0,7021	0,6696

Аналіз отриманих результатів показує стійкість детектора до нових умов. Незначне зниження показників на тестовому наборі даних, у межах 3,5–4% за метриками точної та середньої точності, є закономірним результатом для реальних умов експлуатації та підтверджує здатність моделі до стабільного узагальнення даних.

4.4 Апаратно-програмна оптимізація та низькорівнева квантизація моделі детекції

Розгортання розробленого детектора YOLO26n на бортових ресурсно-обмежених обчислювачах БПЛА-перехоплювачів вимагає радикального

зниження апаратної затримки та оптимізації споживання оперативної пам'яті. За замовчуванням ваги моделі після етапу донавчання експортуються у 32-бітному форматі з плаваючою комою, пряме використання якого на вбудованих платформах призводить до обчислювальних заторів та критичного падіння частоти обробки відеопотоку.

З метою досягнення цільової швидкодії понад 30 кадрів за секунду в реальному часі було спроєктовано та реалізовано конвеєр апаратно-програмної оптимізації моделі YOLO26n. Цей процес включає трансформацію обчислювального графа з формату PyTorch, його низькорівневе стиснення та компіляцію у бінарний рушій інференсу NVIDIA TensorRT (програмна реалізація наведена у додатку Г).

4.4.1 Порівняльний аналіз PyTorch FP32 та TensorRT FP16

На першому етапі оптимізації було здійснено низькорівневу компіляцію моделі YOLO26n у формат напівточності FP16 (програмну реалізацію алгоритмів низькорівневої компіляції та апаратного прискорення в середовищі TensorRT наведено у додатку Ж). Компіляція виконувалася за допомогою інструментарію TensorRT на графічному прискорювачі Tesla T4. Процес передбачав оптимізацію математичного графа, злиття послідовних операцій та автоналаштування ядер під конкретну апаратну архітектуру.

Для оцінки ефективності оптимізації було проведено експериментальне порівняння базової моделі PyTorch FP32 та скомпільованого рушія TensorRT FP16 на тестовій вибірці з 2524 зображень при статичній роздільній здатності 1280×1280 пікселів. Результати порівняння наведено в таблиці 4.4.

Аналіз отриманих емпіричних результатів свідчить про високу ефективність низькорівневої компіляції. Перехід від 32-бітного до 16-бітного представлення ваг практично не вплинув на якість детекції БПЛА.

Метрика mAP_{50} на незалежному тестовому наборі зазнала мізерної деградації – всього на 0.01% абсолютних (з 0.8875 до 0.8874). Це підтверджує, що динамічний діапазон формату FP16 є повністю достатнім для збереження тонких просторових градієнтів та дрібних геометричних ознак малорозмірних повітряних цілей.

Таблиця 4.4 – Порівняльний аналіз базової та квантованої моделей детекції YOLO26n (Kaggle Tesla T4)

Формат представлення моделі	Обсяг ваг, МБ	Середня затримка інференсу, мс	Чиста швидкодія графа, FPS	Метрика mAP_{50}
FP32 PyTorch	5.2	17.93 ± 0.55	55.8	0.8875
TRT FP16	6.9	10.45 ± 1.00	95.7	0.8874

Критично важливою умовою збереження точності моделі під час інференсу на скомпільованому TensorRT-рушії стало примусове вимкнення прямокутного режиму обробки (rect=False) під час валідації. Оскільки бінарний файл .engine будується під суворо фіксовану форму вхідного тензора ($1 \times 3 \times 1280 \times 1280$ BCHW), будь-які спроби динамічно змінювати пропорції кадру на вході призводять до геометричного зміщення та деформації передбачених обмежувальних рамок. Використання стандартного квадратного заповнення повністю нівелювало цей ефект. Середній час чистого виконання графа на GPU знизився з 17.93 мс до 10.45 мс. Це дозволило підняти чисту швидкодію моделі з 55.8 FPS до 95.7 FPS, що створює значний обчислювальний резерв для функціонування алгоритмів темпорального супроводження.

Незважаючи на зниження розрядності чисел, розмір скомпільованого файлу .engine зріс з 5.2 МБ до 6.9 МБ. Це є природним наслідком серіалізації бінарного плану TensorRT, який, окрім самих ваг, містить у собі апаратні

інструкції, метадані обчислювального графа та оптимізовані профілі виконання ядер для конкретного графічного процесора.

4.4.2 Аналіз накладних витрат препроцесингу та рівнів швидкодії

Під час інтеграції оптимізованої моделі у реальний програмний конвеєр обробки відеопотоку швидкодія системи визначається не лише чистим часом роботи GPU, а й накладними витратами на підготовку даних. В межах дослідження було зафіксовано три різні рівні швидкодії моделі TensorRT FP16, що дозволило кількісно оцінити вплив CPU-обмежень на загальну затримку системи:

– чиста швидкодія графа (95,7 FPS при затримці 10,45 мс). Цей показник вимірювався на попередньо згенерованому у відеопам'яті фіктивному масиві випадкових чисел. Він відображає граничну обчислювальну здатність скомпільованих згорткових шарів на тензорних ядрах GPU без залучення інтерпретатора Python та операцій вводу-виводу;

– швидкодія з оптимізованим препроцесингом (98,2 FPS при затримці 10,19 мс). Цей показник отримано під час роботи з реальним тестовим кадром, який був попередньо масштабований до цільового розміру 1280×1280 пікселів за допомогою низькорівневих C++ алгоритмів бібліотеки OpenCV (cv2.resize). Мізерна різниця у 0.26 мс порівняно з чистим графом доводить, що за умови перенесення операцій препроцесингу на скомпільовані C++ бібліотеки накладні витрати CPU є нехтувально малими;

– швидкодія з передобробкою у Python-конвеєрі (79,6 FPS при затримці 12,57 мс) Цей показник відображає найбільш реалістичний сценарій, коли первинне необроблене зображення передається безпосередньо у високорівневу обгортку Ultralytics. Затримка інференсу зросла на 2,38 мс на кадр, що призвело до падіння швидкодії до 79,6 FPS. Ці додаткові 2,38 мс є наслідком роботи інтерпретатора Python на процесорі,

який витрачає час на обчислення пропорцій та додавання бічних полів, конвертацію кольорового простору з BGR в RGB, нормалізацію значень пікселів та перенесення сформованого тензора з системної оперативної пам'яті в пам'ять відеокарти.

4.4.3 Системні обмеження та обґрунтування відмови від INT8-квантизації

Незважаючи на високу теоретичну привабливість переходу до цілочисельного формату низької розрядності INT8, спроба реалізації Post-Training Quantization за допомогою інструментарію NVIDIA ModelOpt завершилася невдачею. Процес компіляції моделі YOLO26n у формат INT8 призвів до критичного збою через два основні чинники:

– графічний прискорювач Tesla T4 базується на мікроархітектурі Turing, яка має вкрай суворі вимоги до вирівнювання та форматів даних для виконання INT8-обчислень на тензорних ядрах. За наявності в графі моделі операцій, які не підтримуються апаратно в INT8 на цій мікроархітектурі, компілятор TensorRT змушений виконувати постійне скидання обчислень. Це призводить до деквантизації тензорів з INT8 назад у FP16/FP32 для виконання операції на звичайних CUDA-ядрах, а потім – до повторного квантування результату в INT8 для наступного шару. Цей безперервний процес перетворення форматів у реєстрах повністю нівелює апаратний приріст швидкодії, роблячи INT8-інференс повільнішим за FP16;

– модель YOLO26n функціонує без традиційного модуля Non-Maximum Suppression, через що її вихідна голова містить велику кількість складних некоординатних операцій транспонування та просторової трансформації тензорів. Інтеграція вузлів квантизації та деквантизації у ці ділянки графу призвела до фрагментації обчислювального графа та унеможливила процес злиття математичних ядер. Компілятор не зміг об'єднати послідовні згортки та активації в єдину інструкцію, що

спричинило помилки типізації при автоматичному виведенні типів даних та падіння загальної продуктивності.

4.4.4 Методологічні обмеження симуляції та екстраполяції на

Отримані в межах симуляції на хмарному прискорювачі Tesla T4 показники швидкодії та затримок є цінним інженерним орієнтиром. Проте безпосередня математична екстраполяція цих цифр на малопотужні вбудовані обчислювачі є некоректною. Продуктивність системи на реальному мобільному пристрої формується під впливом кардинальних відмінностей в апаратній архітектурі.

Серверні відеокарти працюють із власною виділеною високошвидкісною пам'яттю. Натомість компактні вбудовані плати використовують спільну оперативну пам'ять для процесора та графічного ядра. Через це виникає конкуренція за шину даних, коли центральний процесор зчитує кадри з камери, а графічний процесор одночасно намагається зчитувати ваги моделі, що створює додаткові нелінійні затримки обчислень.

Серверні графічні процесори мають значно більшу кількість ядер та іншу мікроархітектуру. Ефективність виконання математичних операцій та підтримка низькорівневих інструкцій суттєво відрізняються залежно від класу та покоління заліза, що робить пряме порівняння теоретичної потужності некоректним.

У хмарному середовищі операції підготовки кадрів та розрахунки трекера виконуються на надпотужних серверних процесорах. На реальному автономному пристрої ці завдання покладаються на енергоефективні мобільні процесори, однопотокова швидкість яких є значно нижчою, що призводить до помітного зростання затримок препроцесингу.

Серверне обладнання працює в кондиційованих приміщеннях зі стабільним живленням. Малопотужні мобільні пристрої обмежені

жорсткими лімітами енергоспоживання та пасивним охолодженням. При тривалому навантаженні вбудований контролер автоматично знижуватиме тактові частоти процесорів для запобігання перегріву, що викликає коливання швидкодії.

Таким чином, результати симуляції у хмарі підтверджують сумісність розробленої оптимізованої моделі з технологіями прискорення та її здатність функціонувати у режимі реального часу. Проте точні показники часу обробки відеосигналу та реальну швидкість роботи всієї системи можна визначити лише під час експериментальної перевірки на фізичному обладнанні (програмна реалізація наведена у додатку Д).

4.5 Аналіз хибнопозитивних та хибнонегативних спрацювань у розрізі семантичних доменів

Загальні інтегральні метрики оцінки ефективності детектора є усередненими показниками і не дозволяють повноцінно оцінити поведінку системи в граничних експлуатаційних умовах. Оскільки навчальна та тестова вибірки спеціально формувалися з урахуванням високого семантичного різноманіття, виникає необхідність проведення детального діагностичного аналізу детектора у розрізі окремих фізичних доменів: геометричних форм цілей, типів камер, складності підстильної поверхні та лінійних розмірів об'єктів детекції.

Базовим кроком для проведення такого дослідження є вибір оптимального порогу спрацьовування детектора, оскільки встановлення межі впевненості для генерації обмежувальних рамок є критичним етапом налаштування контуру управління перехоплювача. Зважаючи на специфіку бойового застосування, архітектура системи вимагає розмежування аналітичних та операційних порогів. Це обґрунтовує необхідність незалежного калібрування детектора за двома метриками: збалансованою інтегральною оцінкою F1 та орієнтованою на точність оцінкою F0.5.

Результати оцінювання моделі YOLO26n за цими двома критеріями зведено в таблицю 4.5.

Таблиця 4.5 – Порівняння ефективності детектора за аналітичним та операційним порогами

Показник ефективності	Аналітичний поріг (Max F1 = 0.35)	Операційний поріг (Max F0.5 = 0.70)
Точність	0.8982	0.9622
Повнота	0.8084	0.7365
F-score	0.8509	0.9066
Рівень хибних спрацювань на негативах	2.47 %	1.65 %
Середня похибка локалізації центроїда	7.36 пікселя	7.36 пікселя

Аналіз даних таблиці 4.5 підтверджує, що математичний максимум метрики F1 досягається за порогу детекції 0.35, надаючи рівноцінну вагу хибнопозитивним та хибнонегативним спрацюванням. Цей показник використовується як інструмент об'єктивного виявлення фундаментальних фізичних обмежень моделі детекції. Проте використання такого порогу для реального польоту є неприпустимим. В умовах автономного самонаведення ціна хибного спрацювання значно перевищує ціну тимчасового пропуску цілі, оскільки захоплення наземного артефакта замість ворожого безпілотника призведе до зриву місії.

Для розв'язання цієї проблеми операційний поріг було відкалібровано за метрикою F0.5, яка математично штрафує систему за хибні спрацювання вдвічі сильніше. Оптимум цієї функції фіксується на рівні 0.70. Застосування цього жорсткого порогу забезпечує екстремально високу точність (0.9622) та знижує частоту хибних тривог до 1.65 %. Висока надійність локалізації додатково підтверджується тим, що середня похибка визначення центроїда цілі становить лише 7.36 пікселя при роздільній здатності кадру 1280x1280.

Для оцінки завадостійкості розробленого детектора та визначення його схильності до генерації помилкових спрацювань на нанегативних зразках було проведено інференс на ізольованій підмножині негативних кадрів загальним обсягом 243 одиниці. За операційного порогу детекції 0,70 модель згенерувала всього 4 помилкові обмежувальні рамки, що відповідає середній частоті помилкових тригерів на рівні 0,016 кадру. Результати розподілу цих помилок за фізичними доменами наведено в таблиці 4.6.

Таблиця 4.6 – Розподіл хибних спрацювань на негативній вибірці за доменами

Домен метаданих	Категорія домену	Кількість хибних рамок
Сенсор камери	rgb	1
	thermal	3
Підстильна поверхня	sky	2
	field/horizon	2

Аналіз даних таблиці 4.6 свідчить про високу стійкість розробленої системи до різноманітних фонових завад та оптичних артефактів. Незначна кількість хибних спрацювань у тепловізійному спектрі пов'язана з наявністю локальних градієнтів температур на складних ділянках підстильної поверхні, які за своєю геометрією та контрастом частково нагадували теплові силуети цілей. Отриманий низький рівень хибних тривог на негативних кадрах (всього 4 помилкові рамки на 243 кадри) експериментально підтверджує ефективність запропонованої у розділі 3 методики інтеграції негативних зразків. Це дозволило примусово знизити чутливість моделі до статичних вотермарок та висококонтрастних елементів OSD-телеметрії у кутах кадру, повністю нівелювавши ефект навчання за спрощеними шляхами. Програмна реалізація наведена у додатку Е.

Для детального статистичного оцінювання працездатності детектора було проведено комплексне тестування на верифікованій позитивній

вибірці обсягом 2281 зображення. За операційного порогу детекції 0,70 зафіксовано 1680 успішних детекцій, 601 пропуск цілі та 62 випадкові хибні рамки. Зведені показники точності, повноти та інтегральної оцінки $F_{0,5}$ у розрізі семантичних доменів наведено на рисунку 4.3.

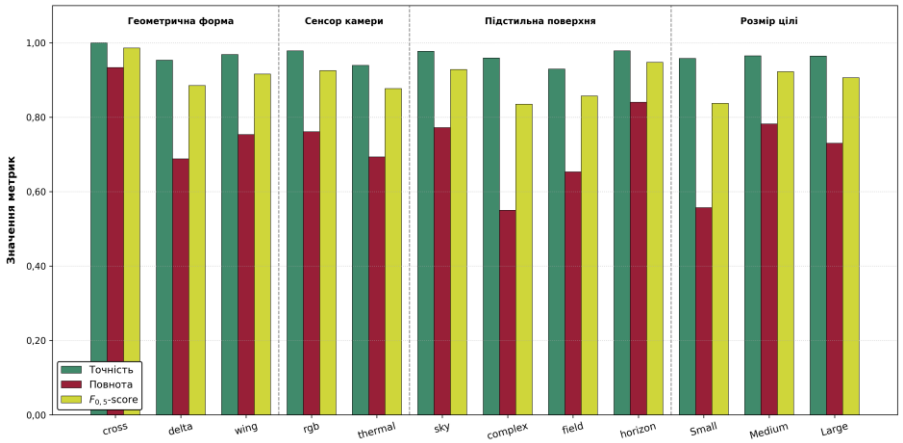


Рисунок 4.3 – Порівняльний аналіз показників ефективності детектора YOLO26n у розрізі семантичних доменів за операційного порогу 0,70

Примітка. Класифікація за розміром цілі відповідає стандарту MS COCO: Small – менше ніж 32 × 32 px; Medium – від 32 × 32 px до 96 × 96 px; Large – понад 96 × 96 px.

Аналіз представлених на рисунку 4.3 результатів дозволяє зробити наступні висновки:

– найвищі показники детекції продемонстрував клас cross ($F_{0,5}$ = 0,986). Його специфічний симетричний X-подібний силует не має природних аналогів у повітряному просторі, що мінімізує ймовірність помилок першого роду. Найнижчий показник повноти зафіксовано для класу delta, чий рівень повноти склав 0,688 за 229 пропусків. На великих дистанціях дельтоподібне крило візуально сприймається як тонка трикутна смуга без вираженого фюзеляжу, що легко маскується градієнтами хмарності або контрастним рельєфом;

– денний оптичний канал демонструє вищу точність порівняно з інфрачервоним. Для тепловізійних знімків показник повноти знижується

до 0,693. Фізичною причиною цього є теплова дифузія та ефект теплового розмиття меж об'єкта, коли нагріті сонячним випромінюванням елементи планера БПЛА починають частково зливатися з розігрітим тлом підстильної поверхні землі.

Стабільно високі результати детекції отримано на лініях динамічного переходу horizon ($F_{0,5}$ = 0,947) та на тлі чистого неба sky ($F_{0,5}$ = 0,928). Істотне зниження повноти детекції спостерігається на складних ландшафтах complex (0,550) та на тлі полів field (0,653). Складні текстури лісових масивів, міської забудови та геометрія доріг створюють високий рівень просторового шуму, який маскує силуети БПЛА та призводить до пропусків цілей.

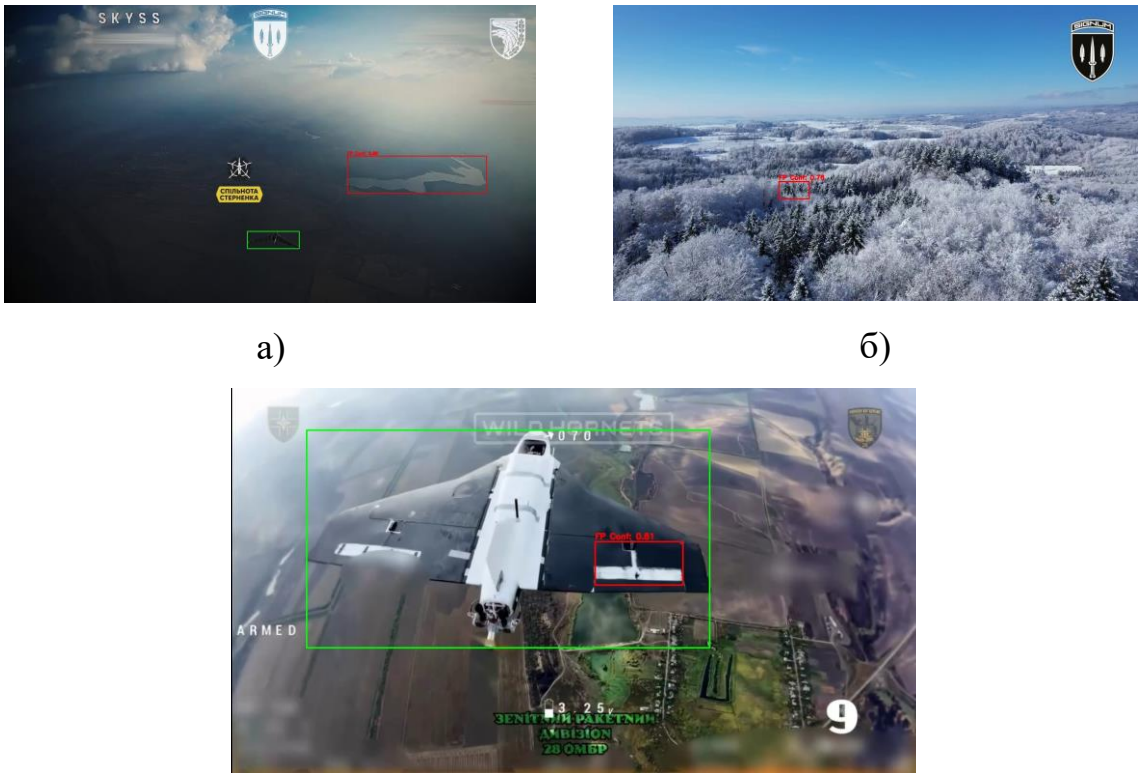
Очікувано нижчий показник повноти зафіксовано для малих мікрооб'єктів (0,557) через втрату високочастотних просторових ознак на великих дистанціях спостереження. Проте виявлено також зниження повноти для великих об’єктів до 0,730 (400 пропусків).

Візуальний аналіз результатів на тестовому наборі даних дозволив детально класифікувати природу помилок детектора, які поділяються на хибні спрацювання та хибнонегативні пропуски цілі.

Хибнопозитивні спрацювання є найбільш критичними для контуру автоматичного наведення, оскільки вони призводять до видачі некоректних кутових координат на автопілот і, як наслідок, до зриву бойової місії. На рисунку 4.3 представлено три типові сценарії, за яких детектор сформував хибні обмежувальні рамки із високим показником впевненості.

Детальний аналіз зафіксованих хибнопозитивних помилок дозволяє зробити кілька важливих висновків. Перша аномалія пов'язана з геометричним зсувом на хмарах, який зображено на рисунку 4.4, а. У цьому випадку модель згенерувала помилкову обмежувальну рамку з рівнем довіри 0.88 на горизонтальній межі купчастої хмарності. Це пояснюється високою концентрацією лінійних градієнтів у цій зоні, які згорткові шари сплутали з довгим та тонким силуетом планера. Натомість справжня

мікроціль у нижній лівій частині кадру, позначена зеленою рамкою, отримала значно менше уваги алгоритму.



а) б)

в)

Рисунок 4.4 – Приклади хибнопозитивних спрацювань детектора YOLO26n за операційного порогу 0.70: а) збій на межі хмарності; б) збій на висококонтрастній ділянці лісового масиву; в) ефект локальної подвійної детекції

Друга проблема стосується контрастного рельєфу підстильної поверхні, представленої на рисунку 4.4, б, де в зимових умовах на засніженому лісовому масиві сталася помилка з рівнем довіри 0.78. Темний силует групи дерев на фоні яскравого снігу утворив специфічний геометричний малюнок, який модель розпізнала як літальний апарат. Це підтверджує, що висококонтрастні природні завади в інфрачервоному або оптичному діапазонах здатні викликати помилкові спрацювання через обмежену роздільну здатність полегшених архітектур. Для усунення

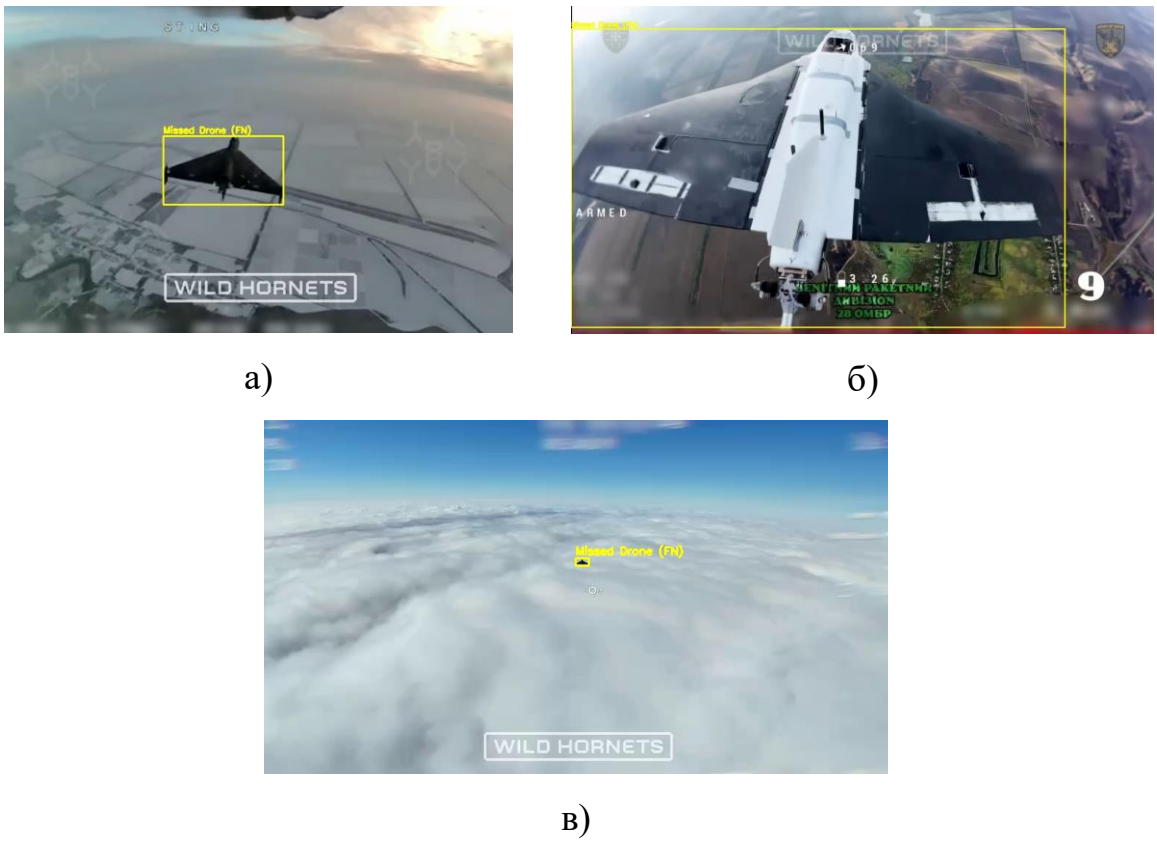
подібних аномалій у майбутніх дослідженнях необхідно застосувати метод цілеспрямованого видобутку складних негативів. Розширення навчальної вибірки масивом зображень специфічних природних та антропогенних артефактів дозволить згортковій мережі сформувати стійкіші дискримінаційні ознаки та суттєво знизити загальний рівень хибних спрацювань.

Третьою аномалією є ефект подвійної детекції окремих частин конструкції, показаний на рисунку 4.4, в. На близькій відстані модель успішно виділила дрон, проте згенерувала додаткову хибну рамку детекції з впевненістю 0.81 на контрастному елементі правого крила. Цей ефект спричинений мультимасштабним конфліктом піраміди ознак. Глибокі шари детектора низької роздільної здатності успішно ідентифікували загальний силует БПЛА, тоді як шари високої роздільної здатності, оптимізовані для дрібних цілей, помилково сприйняли локальну висококонтрастну білу смугу на крилі як окремий віддалений планер. Оскільки архітектура YOLO26 функціонує за принципом прямого виведення координат без застосування NMS алгоритму, вона позбавлена зовнішнього математичного апарату усунення перекриттів. Це зумовлює її вразливість до виникнення локальних подвійних спрацювань за умов суттєвої розбіжності між характеристиками навчальних та реальних даних.

З іншого боку, найбільшою загрозою для безперервності роботи автоматичного супроводжувача є пропуски цілей. За умови встановлення операційного порогу на рівні 0.70 загальна кількість пропусків склала 601 випадок. На рисунку 4.5 наведено характерні сценарії, за яких модель не змогла виявити об'єкт.

На рисунку 4.5, а показано пропуск дельтоподібного безпілотника через його маскування рельєфом землі. Контрастний малюнок полів та доріг створив високий рівень геометричного шуму, який приховав силует об'єкта, через що рівень розпізнавання цілі впав нижче порогу 0.70.

Програмна реалізація наведена у додатку Ж.



а) б)

в)

Рисунок 4.5 – Приклади хибнонегативних пропусків цілей детектора YOLO26n за операційного порогу 0.70: а) маскування складним фоном землі; б) кінематичне розмиття та вихід за межі кадру; в) деградація ознак мікророзмірної цілі на фоні хмар

Іншою аномалією є пропуск великого апарата на близькій відстані (рисунок 4.5, б), що пояснюється обмеженням надлегкої архітектури YOLO26n. Через малий розмір зон сприйняття згорткових ядер модель не здатна охопити геометричні ознаки об'єкта (крила та корпус), коли він займає майже весь кадр розміром 1280 на 1280 точок. Це зумовлено тим, що ультралегкі архітектури класу Nano мають обмежену глибину згорткових шарів, що фізично звужує їхнє максимальне рецептивне поле. Як наслідок, при екстремальному наближенні модель починає аналізувати крило чи фюзеляж як окремі ізольовані структури тла, втрачаючи здатність розуміти цілісний образ БПЛА. У результаті мережа втрачає здатність

розпізнати такий великий об'єкт, а різкий двоколірний розподіл кольору його корпусу додатково руйнує візуальну цілісність цілі.

Крім того, на великій дистанції (рисунок 4.5, в) виникає пропуск через фрагментацію дрібної цілі. Під час проходження кадру крізь глибокі шари моделі YOLO26n фільтри розділяють малий силует на кілька окремих частин. Ці розрізнені фрагменти нівелюються операціями згортки на фоні хмарної текстури, що унеможливлює розпізнавання об'єкта.

Підсумовуючи результати візуального аналізу, необхідно констатувати, що вирішення задачі наведення виключно на рівні покадрової просторової детекції є неповним. Пріоритетним завданням бортового контуру управління є не просто одномоментна фіксація об'єкта, а створення його безперервної кінематичної траєкторії в часі, що є абсолютно необхідною умовою для розрахунку вектора перехоплення автопілотом. Це формує базову архітектурну потребу в інтеграції спеціалізованої підсистеми темпорального трекінгу. Водночас наявність 601 хибнонегативного спрацювання за операційного порогу 0.70 розкриває другу, не менш критичну функцію алгоритмів супроводження: вони діють як потужний математичний компенсатор вразливостей детектора. Завдяки здатності прогнозувати рух об'єкта, трекер успішно реконструює розірвані траєкторії під час неминучих оптичних провалів, нівелюючи покадрові пропуски згорткової мережі та гарантуючи стабільне утримання цілі впродовж усього маневру (програмна реалізація наведена у додатку И).

4.6 Інтеграція та порівняльний аналіз алгоритмів темпорального трекінгу

На завершальному етапі експериментальних досліджень було розгорнуто та оптимізовано підсистему темпорального супроводження (програмний код для моделювання, тестування та оцінювання трекерів наведено у додатку К), яка відповідає за відстеження

траєкторії польоту безпілотних апаратів у реальному часі. Для інтеграції в контур управління безпілотника-перехоплювача було обрано два найбільш перспективні алгоритми, які не потребують значних обчислювальних ресурсів центрального процесора: ByteTrack та OC-SORT. З метою визначення найкращого алгоритму було виконано порівняльне тестування обох систем у двох режимах: спочатку з використанням еталонних міток розмітки, а потім із подачею на вхід передбачень оптимізованої моделі детекції.

Для комплексної оптимізації робочих параметрів системи супроводження було застосовано метод систематичного перебору за сіткою гіперпараметрів (Grid Search) на основі контрольних показників валідаційного набору даних. У процесі калібрування базового алгоритму ByteTrack послідовно змінювалися поріг ініціалізації нових траєкторій, тривалість збереження втраченого треку в оперативній пам'яті, а також поріг остаточної асоціації детекцій. Для алгоритму OC-SORT аналогічно варіювалися поріг фільтрації вхідних детекцій, граничний час життя траєкторії без нових спостережень та поріг перетину обмежувальних рамок. Аналітичні результати перебору дозволили визначити найбільш раціональні робочі налаштування. Для алгоритму ByteTrack оптимальними визначено такі значення: поріг детекції на рівні 0.3, буфер збереження траєкторій на рівні 30 кадрів та поріг зіставлення на рівні 0.9. Для OC-SORT найефективнішою виявилася конфігурація з порогом детекції 0.3, граничним часом життя траєкторії 30 кадрів і порогом перетину обмежувальних рамок (IoU) на рівні 0.2.

Для оцінки теоретичного максимуму стійкості алгоритмів супроводження без урахування випадкових помилок детектора обидва трекери з найкращими оптимізованими параметрами було протестовано з використанням істинної розмітки траєкторій. Показники ефективності цього випробування наведено в таблиці 4.7.

Таблиця 4.7 – Показники ефективності алгоритмів супроводження на істинних мітках

Алгоритм	Поріг детекції	Пам'ять у кадрах	Поріг перетину	HOTA	IDF1	MOTA
ByteTrack	0,30	30	0,90	85,92%	96,23%	97,80%
OC-SORT	0,30	30	0,20	96,87%	96,53%	98,32%

Аналіз отриманих результатів на істинних мітках свідчить про значну математичну перевагу алгоритму OC-SORT. Інтегральна точність відстеження HOTA для OC-SORT досягла високого значення 96.87 %, тоді як для ByteTrack цей показник склав 85.92 %. Показник збереження ідентифікатора траєкторії (IDF1) для обох трекерів знаходиться на стабільно високому рівні (понад 96 %), проте OC-SORT демонструє дещо кращу загальну точність (MOTA = 98.32 %). Така стабільність пояснюється тим, що OC-SORT розроблявся спеціально для компенсації складних маневрів та різких розворотів цілей в умовах нелінійної кінематики, де алгоритм успішно коригує стан фільтра Калмана на основі реальних точок спостережень. Тестування на ідеальній розмітці дозволило оцінити граничний математичний потенціал кожного трекера в ізоляції від помилок нейромережевого детектора. Отримані високі базові значення метрик HOTA та IDF1 експериментально доводять, що кінематичні моделі обох алгоритмів мають достатній рівень апроксимації для опису складної тривимірної траєкторії польоту БПЛА-цілі.

Для перевірки стійкості роботи системи в реальних умовах експлуатації та визначення здатності алгоритмів компенсувати шуми та пропуски детекції, буловедено аналогічне тестування на детекціях, згенерованих фінальною моделлю YOLO26n. Показники цього дослідження наведено в таблиці 4.8.

Таблиця 4.8 – Показники ефективності супроводження на реальних детекціях моделі YOLO26n

Алгоритм	Поріг детекції	Пам'ять у кадрах	Поріг перетину	HOTA	IDF1	MOTA
ByteTrack	0.60	15	0.90	69.26%	84.14%	77.29%
OC-SORT	0.40	60	0.20	61.38%	69.11%	69.99%

Варто зазначити специфіку налаштування порогів впевненості. Якщо на етапі оцінювання інтегральних метрик детектора використовувався мінімальний поріг 0.01 для побудови повної кривої Precision-Recall, то для роботи в контурі темпорального супроводження такий підхід є неприпустимим через ризик перенасичення фільтра Калмана хибнопозитивними спрацюваннями. Відповідно до результатів алгоритму Grid Search, операційні пороги детекції для підсистеми трекінгу були значно підвищені: до 0.60 для алгоритму ByteTrack та 0.40 для OC-SORT. Це дозволило досягти оптимального балансу між фільтрацією фонових артефактів та збереженням корисного сигналу від цілі під час реального відеопотоку.

Під час роботи з реальними відповідями детектора спостерігається кардинальний зсув результатів: алгоритм ByteTrack демонструє очевидну перевагу над OC-SORT за всіма ключовими метриками. Інтегральна точність відстеження HOTA для ByteTrack склала 69.26 %, тоді як для OC-SORT цей показник впав до 61.38 %. Найбільш показовою є динаміка метрики IDF1, яка безпосередньо відображає безперервність супроводження та здатність трекера утримувати єдиний ідентифікатор цілі впродовж усього польоту: ByteTrack зберіг цей показник на рівні 84.14 %, тоді як точність ідентифікації OC-SORT критично деградувала до 69.11 %. Це свідчить про те, що в умовах шумних детекцій OC-SORT значно частіше розриває траєкторії та створює хибні ідентифікатори для одного й того ж безпілотника.

Цей парадокс є логічним наслідком різниці в архітектурі обробки помилок детектора. Реальна модель детекції на складному фоні підстильної поверхні має пропуски дрібних цілей та створює поодинокі хибні спрацювання. У таких умовах OC-SORT починає накопичувати помилки, оскільки він занадто сильно спирається на останні точні вимірювання, і при зникненні детекцій фільтр Калмана втрачає орієнтири, розриваючи трек і створюючи нові траєкторії при повторній появі цілі, що й спричиняє обвал метрики IDF1. Натомість ByteTrack успішно усуває пропуски детектора завдяки двопрохідному алгоритму асоціації. Замість ігнорування рамок із низькою впевненістю, які зазвичай утворюються під час оптичного розмиття БПЛА в польоті, ByteTrack використовує їх на другому кроці зіставлення для відновлення розірваних траєкторій, що дозволяє стабільно утримувати ідентифікатор цілі протягом усього маневру.

Окрім оцінки стійкості супроводження, вирішальним критерієм вибору алгоритму є його обчислювальна складність. Під час тестування було виміряно чисті затримки оновлення стану трекерів у межах бортового контуру самонаведення БПЛА без урахування накладних витрат на кодування відео. Середня затримка оновлення стану на один кадр для ByteTrack склала 0,568 мс, що відповідає швидкодії у 1759 кадрів за секунду. Для OC-SORT цей показник склав 0,577 мс (1731 кадрів за секунду).

Обидва алгоритми є обчислювально легкими та оптимізованими для роботи на центральному процесорі. Це доводить, що під час розгортання системи на вбудованих обчислювачах підсистема темпорального трекінгу практично не створюватиме додаткового навантаження на процесорні ядра, залишаючи понад 98% обчислювальних ресурсів вільними для розрахунку кінематики перехоплення та функціонування польотного контролера. З огляду на значну перевагу у стійкості супроводження реальних цілей на складному фоні та мінімальну кількість критичних помилок ідентифікації,

алгоритм ByteTrack затверджується як оптимальний вибір для інтеграції в контур управління безпілотника-перехоплювача.

4.7 Теоретична оцінка наскрізної швидкодії програмного конвеєра

Для підтвердження життєздатності розробленої архітектури в умовах обмежень реального часу було проведено оцінку теоретичної наскрізної затримки програмного конвеєра. Зважаючи на неможливість безпосереднього апаратного тестування швидкодії всієї системи на вбудованих пристроях в межах цього етапу дослідження, розрахунок базується на підсумовуванні ізольованих метрик, отриманих у середовищі Kaggle.

Повний цикл базової обробки одного кадру в контурі автономного супроводження складається з двох послідовних етапів. Перший етап охоплює препроцесинг зображення та виконання інференсу моделі YOLO26n у квантованому форматі TensorRT FP16, що згідно з результатами попередніх вимірювань займає 12.57 мс. Другий етап полягає у розрахунку матриці асоціацій та оновленні стану фільтра Калмана за алгоритмом ByteTrack, що потребує додаткових 0.57 мс процесорного часу.

Таким чином, сумарна програмна затримка обробки одного кадру становить 13.14 мс. Цей показник забезпечує теоретичну пропускну здатність конвеєра на рівні 76 кадрів за секунду.

Стандартна частота оновлення відеопотоку з курсових камер перехоплювачів становить 30 кадрів за секунду, що формує жорсткий часовий бюджет у 33.3 мс на обробку одного кадру. Отримана на тестовому стенді затримка у 13.14 мс свідчить про те, що розроблена система використовує менше сорока відсотків доступного часу.

Як було обґрунтовано в підрозділі 4.4, пряма екстраполяція цих метрик на вбудовані обчислювальні платформи є неможливою через суттєві відмінності в пропускній здатності спільної пам'яті та продуктивності

мобільних процесорів. Проте наявність двократного обчислювального резерву (понад 20 мс вільного часу на кожен кадр) дозволяє надійно компенсувати очікуване зниження швидкодії під час розгортання системи на бортовому комп'ютері та забезпечить стабільну видачу координат цілі для автопілота в режимі жорсткого реального часу. Окрім цього, подальше перенесення обчислень на вбудовані пристрої відкриває можливість успішного проведення цілочисельного квантування моделі у формат INT8, що забезпечить додаткове зростання продуктивності системи та знизить апаратні затримки.

У результаті експериментальних досліджень вирішено проблему виникнення явища навчання за спрощеними шляхами. Завдяки аналізу карт активації EigenCAM доведено, що модель без негативних зразків схильна орієнтуватися на водяні знаки та OSD-елементи у кутах кадру. Інтеграція до навчального набору негативних прикладів дозволила переключити увагу глибоких шарів YOLO26n на пошук морфології БПЛА в центральній частині кадру. Зважаючи на жорсткі вимоги системи напівавтоматичного самонавдення до надійності захоплення цілі, операційний поріг детектора було відкалібровано за метрикою $F_{0,5}$. Оптимум цієї функції зафіксовано на рівні 0,70, що забезпечило точність розпізнавання 0,9622, знизило рівень хибних тривог на негативах до 1,65 % та дозволило визначати координати центроїда цілі з середньою похибкою у 7,36 пікселя.

Низькорівнева компіляція та квантизація моделі у формат напівточності TensorRT FP16 на графічному прискорювачі Tesla T4 дозволили знизити середню затримку інференсу з 17,93 мс до 10,45 мс при деградації точності детекції на рівні 0,01 % абсолютних за метрикою mAP_{50}. Спроба застосувати INT8-квантування виявилася невдалою через вимушений апаратний відкат обчислень на архітектурі Turing та порушення топології компіляції шарів YOLO26n, що функціонують без NMS-модуля.

Під час порівняльного аналізу алгоритмів темпорального супроводження в умовах шумних детекцій експериментально доведено перевагу ByteTrack над OC-SORT. Завдяки двопрохідній асоціації BYTE трекер ByteTrack компенсував покадрові пропуски моделі, продемонструвавши інтегральну точність HOTA на рівні 69,26 % та стабільність ідентифікатора IDF1 на рівні 84,14 %, тоді як OC-SORT через чутливість до помилок детектора показав 61,38 % HOTA та 69,11 % IDF1.

Сумарна обчислена затримка наскрізного контуру обробки склала 13,14 мс, що еквівалентно пропускній здатності у 76 FPS. Це підтверджує апаратну життєздатність розробленої системи для її подальшого розгортання на вбудованому обладнанні.

ВИСНОВКИ

У кваліфікаційній роботі виконано проєктування, розробку та експериментальне дослідження програмного конвеєра комп'ютерного зору реального часу для системи напівавтономного повітряного перехоплення безпілотних літальних апаратів. На основі проведеного аналізу специфіки предметної галузі було сформульовано жорсткі тактико-технічні вимоги до бортового обчислювача системи перехоплення. Математично доведено, що для запобігання втраті цілі через високі швидкості зближення мінімальна частота обробки відеопотоку має складати не менше 30 FPS. Це дозволило визначити граничні рамки для проєктування системи в умовах жорстких апаратних обмежень бортового обладнання, зокрема ліміту енергоспоживання до 15 Вт та пасивного охолодження.

Для створення навчальної вибірки було розроблено та практично реалізовано автоматизований конвеєр збору, семантичного маркування та каталогізації первинних OSINT-відеоматеріалів бойової роботи. Завдяки інтеграції мультимодальної мовної моделі Gemini 2.5 Pro через хмарний інтерфейс Vertex AI API, було повністю автоматизовано процес класифікації та кодування метаданих. Для прискорення розмітки кадрів впроваджено гібридну концепцію анотування Human-in-the-Loop, що поєднує автоматичну генерацію початкових псевдоанотацій та трекінг об'єктів моделлю SAM 3 з наступним ручним коригуванням.

Усунення часової та візуальної надмірності відеопотоку було здійснено шляхом впровадження триетапного алгоритму дедуплікації кадрів, який поєднує просторовий аналіз за метрикою IoU та семантичне групування на основі зважених ембедингів моделі DINOv2 за допомогою алгоритму k-Center Greedy.

Для запобігання явищу навчання за спрощеними шляхами розроблено методологію інтеграції синтетичних негативних зразків із накладеними водяними знаками, що дозволило сформувати фінальний дедуплікований

датасет загальним обсягом 13 200 кадрів (12 000 позитивних та 1 200 негативних зразків).

Експериментальна оцінка ефективності запропонованої методології підготовки даних проводилася шляхом навчання моделі YOLO26n. Аналізом EigenCAM доведено, що інтеграція негативних зразків успішно переключила увагу моделі зі статичної графіки безпосередньо на морфологічні ознаки БПЛА. Оптимізована модель продемонструвала високу точність детекції (mAP50 = 0,9094 на валідації та mAP50 = 0,8668 на незалежному тестовому наборі). Калібрування операційного порогу за метрикою $F_{0.5}$ до рівня 0,70 забезпечило високу точність (0,9622) та знизило частоту хибних тривог на негативах до 1,65%.

Низькорівнева компіляція та квантизація моделі у формат напівточності TensorRT FP16 на графічному прискорювачі Tesla T4 дозволили знизити середню затримку інференсу з 17,93 мс до 10,45 мс, що еквівалентно 95,7 FPS, при мізерній деградації точності на рівні 0,01% mAP50. Спроба застосувати INT8-квантування виявилася невдалою через вимушений апаратний відкат обчислень на мікроархітектурі Turing та порушення топологии компіляції шарів YOLO26n, що функціонують без NMS-модуля. Водночас під час порівняльного аналізу алгоритмів темпорального супроводження в умовах шумних детекцій експериментально доведено перевагу ByteTrack над OC-SORT (інтегральна точність HOTA склала 69,26%, стабільність ідентифікатора IDF1 – 84,14%) при середній затримці оновлення стану всього 0,568 мс на кадр.

Сумарна обчислена затримка наскрізного контуру обробки склала 13,14 мс, що еквівалентно пропускній здатності у 76 FPS. Наявність двократного обчислювального резерву понад 20 мс вільного часу на кожен кадр відносно ліміту в 33,3 мс дозволяє надійно компенсувати очікуване зниження швидкодії при розгортанні системи на бортовому комп'ютері та надійно забезпечить стабільну видачу координат цілі для автопілота в режимі жорсткого реального часу на цільовому вбудованому обладнанні.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Шумлянський Д. Реактивний «Шахед»: не дешева ерзац-крилата ракета та нова загроза для України. *Мілітарний*. URL: https://militarnyi.com/uk/articles/reaktyvnyj-shahed-ne-desheva-erzats-krylata-raketa-ta-nova-zagroza-dlya-ukrayiny/ (дата звернення: 19.05.2026).

2. Security. Безпілотна авіаційна платформа «X2 Jet». *Brave1 Market – технології для ЗСУ від українських виробників*. URL: https://market-brave1.delta.mil.gov.ua/kopter/6305/ (дата звернення: 19.05.2026).

3. Security. Багатоцільовий безпілотний комплекс «Багнет-АА». *Brave1 Market – технології для ЗСУ від українських виробників*. URL: https://market-brave1.delta.mil.gov.ua/kopter/6691/ (дата звернення: 19.05.2026).

4. Benchmark analysis of YOLO performance on edge intelligence devices / H. Feng et al. Cryptography. 2022. Vol. 6, no. 2. P. 16. URL: https://doi.org/10.3390/cryptography6020016 (date of access: 19.05.2026).

5. Bamberg L. EIQ neutron: redefining edge-ai inference with integrated NPU and compiler innovations. *arXiv.org*. URL: https://arxiv.org/abs/2509.14388 (date of access: 19.05.2026).

6. Rockchip RK3588S Datasheet. *Rockchip TRM and Datasheets*. URL: https://rockchip.fr/RK3588S%20datasheet%20V1.7.pdf (date of access: 19.05.2026).

7. Karimov T. Quantization robustness to input degradations for object detection. *arXiv.org*. URL: https://doi.org/10.48550/arXiv.2508.19600 (date of access: 19.05.2026).

8. Ultralytics YOLO26 | ultralytics docs. *Ultralytics Docs*. URL: https://docs.ultralytics.com/models/yolo26/#detection-coco (date of access: 19.05.2026).

9. Zhang Y. ByteTrack: multi-object tracking by associating every detection box. *arXiv.org*. URL: https://arxiv.org/abs/2110.06864 (date of access: 19.05.2026).

10. Ultralytics YOLO11 | ultralytics docs. *Ultralytics Docs*. URL: https://docs.ultralytics.com/models/yolo11 (date of access: 19.05.2026).

11. Cao J. Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking. *arXiv.org*. URL: https://arxiv.org/abs/2203.14360 (date of access: 19.05.2026).

12. VisDrone-DET2019: the vision meets drone object detection in image challenge results / D. Du et al. *2019 IEEE/CVF international conference on computer vision workshop (ICCVW)*, Seoul, Korea (South), 27–28 October 2019. 2019. URL: https://doi.org/10.1109/iccvw.2019.00030 (date of access: 19.05.2026).

13. Jiang N. Anti-UAV: a large multi-modal benchmark for UAV tracking. *arXiv.org*. URL: https://arxiv.org/abs/2101.08466 (date of access: 19.05.2026).

14. Drone vs. bird detection: deep learning algorithms and results from a grand challenge / A. Coluccia et al. *Sensors*. 2021. Vol. 21, no. 8. P. 2824. URL: https://doi.org/10.3390/s21082824 (date of access: 19.05.2026).

15. VisioDECT: a robust dataset for aerial and scenario based multi-drone detection, identification, and neutralization / S. O. Ajakwe та ін. *Data in brief*. 2026. P. 112448. URL: https://doi.org/10.1016/j.dib.2026.112448 (date of access: 19.05.2026).

16. Chen P. SDD-YOLO: a small-target detection framework for ground-to-air anti-UAV surveillance with edge-efficient deployment. *arXiv.org*. URL: https://doi.org/10.48550/arXiv.2603.25218 (date of access: 07.07.2026).

17. Tips for best yolov5 training results | ultralytics docs. *Ultralytics Docs*. URL: https://docs.ultralytics.com/yolov5/tutorials/tips_for_best_training_results (date of access: 19.05.2026).

18. Ye W. Spurious correlations in machine learning: a survey. *arXiv.org*. URL: https://arxiv.org/abs/2402.12715v1 (date of access: 19.05.2026).

19. Lapuschkin S. Unmasking clever hans predictors and assessing what machines really learn. *arXiv.org*. URL: https://arxiv.org/abs/1902.10178 (date of access: 19.05.2026).

20. Wang S. Do ImageNet-trained models learn shortcuts? The impact of frequency shortcuts on generalization. *arXiv.org*. URL: https://arxiv.org/abs/2503.03519 (date of access: 19.05.2026).

21. DeVries T. Improved regularization of convolutional neural networks with cutout. *arXiv.org*. URL: https://arxiv.org/abs/1708.04552 (date of access: 19.05.2026).

22. Ahmed K. R. FryNet: dual-stream adversarial fusion for non-destructive frying oil oxidation assessment. *arXiv.org*. URL: https://arxiv.org/abs/2604.21321 (date of access: 19.05.2026).

23. Liu C.-M., Juang J.-C. Systematic evaluation of yolov8 variants for uav-based object detection. *Applied sciences*. 2026. Vol. 16, no. 7. P. 3559. URL: https://doi.org/10.3390/app16073559 (date of access: 19.05.2026).

24. Lin G. DUST-YOLO: an efficient and lightweight deployable UAV swin transformer YOLO for end-to-end video object detection. *Preprints*. 2026. URL: https://doi.org/10.20944/preprints202604.1422.v1 (date of access: 19.05.2026).

25. Small-Object detection at the edge: A pareto-efficient benchmark of lightweight YOLO models on UAV and overhead datasets / B. Shakya та ін. *IEEE access*. 2025. P. 1. URL: https://doi.org/10.1109/access.2025.3648968 (date of access: 19.05.2026).

26. Karjol A. Edge AI for automotive vulnerable road user safety: deployable detection via knowledge distillation. *arXiv.org*. URL: https://arxiv.org/abs/2604.26857 (date of access: 19.05.2026).