

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)

\_\_\_\_\_ Методи визначення, класифікації та прогнозування рухомих об'єктів у  
\_\_\_\_\_ повітряному просторі на основі глибокого навчання  
(тема)

Виконав:  
здобувач \_\_\_\_\_ четвертого \_\_\_\_\_ року навчання,  
групи \_\_\_\_\_ ІТШ-21-2

\_\_\_\_\_ Оксана Христенко  
(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. Олена Волощук  
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ \_\_\_\_\_  
(підпис)

\_\_\_\_\_ Олег ЗОЛОТУХІН  
(власне ім'я, прізвище)

2025 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 \_\_\_\_\_ Христенко Оксані Владиславівні \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи \_\_\_\_\_ Методи визначення, класифікації та прогнозування рухомих об'єктів у повітряному просторі на основі глибокого навчання \_\_\_\_\_

затверджена наказом університету від 19 травня 2025 р. № 378Ст

2. Термін подання студентом роботи до екзаменаційної комісії 18 червня 2025 р.

3. Вихідні дані до роботи \_\_\_\_\_ Науково-технічні публікації, дані Інтернет-джерел та наукових проектів щодо розробки та дослідження систем виявлення та відстеження об'єктів, нейронних мереж YOLO, фільтрів Калмана, набори даних з відео дронів та птахів, документація Python, OpenCV, PyTorch, Ultralytics \_\_\_\_\_

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

1) Аналіз предметної галузі та теоретичні основи відстеження об'єктів у повітряному просторі

2) Методи визначення, класифікації та прогнозування руху дронів

3) Реалізація та експериментальна оцінка системи відстеження дронів



## РЕФЕРАТ

Пояснювальна записка: 83 с., 16 рис., 13 табл., 1 дод., 29 джерел.

БЕЗПЛОТНІ ЛІТАЛЬНІ АПАРАТИ, ВІДСТЕЖЕННЯ ОБ'ЄКТІВ, ГЛИБИННЕ НАВЧАННЯ, ДРОНИ, КЛАСИФІКАЦІЯ ОБ'ЄКТІВ, КОМП'ЮТЕРНИЙ ЗІР, НЕЙРОННІ МЕРЕЖІ, ПОВІТРЯНИЙ ПРОСТІР, ТРЕКІНГ, ФІЛЬТР КАЛМАНА, YOLO.

Об'єкт дослідження – процес відстеження та прогнозування руху об'єктів у повітряному просторі.

Предмет дослідження – методи визначення, класифікації та прогнозування руху БПЛА на основі комп'ютерного зору та глибокого навчання.

Мета роботи – розробка системи для детекції, трекінгу та прогнозування траєкторії руху дронів, яка ефективно розрізняє дрони від птахів.

Методи дослідження – теоретичний (аналіз існуючих підходів), експериментальний (розробка архітектури, навчання моделей), математичний (фільтрація траєкторій, статистичний аналіз).

Створено комплексну систему детекції, трекінгу та прогнозування руху дронів на основі YOLOv8 та фільтра Калмана. Система демонструє mAP50 = 0.914, F1-score 0.92 для розрізнення дронів від птахів та здатність працювати в реальному часі.

Практична цінність полягає у розробці системи для моніторингу повітряного простору, забезпечення безпеки критичної інфраструктури та контролю заборонених зон польотів БПЛА. Система функціонує на доступному обладнанні та інтегрується з існуючими системами безпеки.

Ключове досягнення – ефективне розрізнення дронів від птахів, що мінімізує помилкові спрацьовування систем безпеки.

## ABSTRACT

Bachelor's thesis contains: 83 pp., 16 fig., 13 tabl., 1 ann., 29 references.

AIRSPACE, COMPUTER VISION, DEEP LEARNING, DRONES, KALMAN FILTER, NEURAL NETWORKS, OBJECT CLASSIFICATION, OBJECT TRACKING, TRACKING, UNMANNED AERIAL VEHICLES, YOLO.

Object of research – the process of tracking and predicting the movement of objects in airspace.

Subject of research – methods for detection, classification, and prediction of UAV movement based on computer vision and deep learning.

Purpose of the work – development of a system for detection, tracking, and trajectory prediction of drone movement that effectively distinguishes drones from birds.

Research methods – theoretical (analysis of existing approaches), experimental (architecture development, model training), mathematical (trajectory filtering, statistical analysis).

A comprehensive system for drone detection, tracking, and movement prediction based on YOLOv8 and Kalman filter was created. The system demonstrates  $mAP50 = 0.914$ , F1-score 0.92 for distinguishing drones from birds, and real-time operation capability.

Practical value lies in developing a system for airspace monitoring, ensuring security of critical infrastructure, and controlling prohibited UAV flight zones. The system operates on affordable equipment and integrates with existing security systems.

Key achievement – effective distinction between drones and birds, which minimizes false alarms in security systems.

## ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів .....	7
Вступ.....	9
1 Аналіз предметної галузі та теоретичні основи відстеження об'єктів у повітряному просторі.....	11
1.1 Актуальність відстеження та прогнозування руху об'єктів у повітряному просторі.....	11
1.2 Аналіз предметної галузі.....	13
1.3. Технологічні аспекти побудови систем відстеження.....	17
1.4 Теоретичні основи визначення та класифікації рухомих об'єктів .....	19
1.5 Постановка завдання дослідження .....	25
2 Методи визначення, класифікації та прогнозування руху дронів .....	29
2.1 Обґрунтування методів дослідження .....	29
2.2 Методи виявлення та класифікації повітряних об'єктів.....	32
2.3 Методи трекінгу та прогнозування траєкторії.....	34
2.4 Методика тестування та оцінки ефективності.....	41
3 Реалізація та експериментальна оцінка системи відстеження дронів .....	45
3.1 Розробка архітектури системи відстеження дронів.....	45
3.2 Реалізація модулів підготовки та обробки даних .....	48
3.3 Імплементация моделі виявлення та класифікації об'єктів.....	52
3.4 Розробка системи трекінгу з використанням фільтра Калмана .....	60
3.5 Алгоритми прогнозування траєкторії руху.....	67
3.6 Шляхи подальшого вдосконалення системи .....	74
Висновки.....	77
Перелік джерел посилання .....	79
Додаток А Відомість кваліфікаційної роботи .....	83

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

- БПЛА – безпілотний літальний апарат;
- CA – Constant Acceleration – модель з постійним прискоренням;
- CSV – Comma-Separated Values – значення, розділені комами (формат файлу);
- CUDA – Compute Unified Device Architecture – архітектура паралельних обчислень від NVIDIA;
- CV – Constant Velocity – модель з постійною швидкістю;
- F1-score – гармонічне середнє між точністю та повнотою;
- FPS – Frames Per Second – кадрів за секунду;
- GPU – Graphics Processing Unit – графічний процесор;
- IoU – Intersection over Union – метрика перетину над об'єднанням;
- JSON – JavaScript Object Notation – формат обміну даними;
- MAE – Mean Absolute Error – середня абсолютна помилка;
- mAP – mean Average Precision – середня точність;
- mAP50 – mean Average Precision при порозі IoU=0.5;
- MOTA – Multiple Object Tracking Accuracy – точність відстеження множинних об'єктів;
- MOTP – Multiple Object Tracking Precision – точність локалізації при відстеженні;
- NMS – Non-Maximum Suppression – пригнічення не-максимумів;
- OpenCV – Open Source Computer Vision Library – бібліотека комп'ютерного зору;
- Precision – точність (частка правильних виявлень серед усіх виявлень);
- PyTorch – бібліотека для машинного навчання;
- Recall – повнота (частка виявлених об'єктів серед наявних);
- RMSE – Root Mean Square Error – середньоквадратична помилка;
- ROI – Region of Interest – область інтересу;

SORT – Simple Online and Realtime Tracking – алгоритм відстеження об'єктів;

TensorRT – бібліотека для оптимізації глибоких нейронних мереж;

YAML – YAML Ain't Markup Language – формат серіалізації даних;

YOLO – You Only Look Once – алгоритм детекції об'єктів.

## ВСТУП

Моніторинг повітряного простору набуває все більшого значення в сучасному світі, де повітряний рух стає інтенсивнішим і різноманітнішим. Окрім традиційних пілотованих літальних апаратів, стрімко зростає кількість безпілотних літальних апаратів (БПЛА) різного розміру та призначення, що створює принципово нові виклики для систем спостереження та безпеки.

Актуальність розробки ефективних методів визначення, класифікації та прогнозування руху об'єктів у повітряному просторі зумовлена низкою факторів. По-перше, доступність комерційних дронів призвела до їх масового використання у різноманітних сферах – від розваг та медіа до логістики й сільського господарства. Це супроводжується зростанням кількості інцидентів, пов'язаних із несанкціонованим використанням БПЛА у заборонених зонах, особливо поблизу аеропортів, критичної інфраструктури та місць проведення масових заходів. По-друге, сучасні дрони стають менш помітними, тихішими та здатними переносити небезпечні вантажі, що створює додаткові загрози для безпеки. По-третє, перехід від централізованих систем управління повітряним рухом до розподілених мереж, що інтегрують пілотовані та безпілотні апарати, вимагає нових підходів до відстеження та контролю.

Особливої уваги заслуговує проблема розрізнення дронів та птахів, які можуть мати подібні розміри, швидкості та характеристики руху, особливо на значній відстані або в умовах обмеженої видимості. Традиційні системи радіолокаційного спостереження часто не забезпечують достатньої роздільної здатності для надійної ідентифікації малорозмірних об'єктів, а оптичні системи стикаються з обмеженнями, пов'язаними з погодними умовами та освітленням.

У цьому контексті застосування методів глибокого навчання відкриває нові можливості для створення ефективних систем моніторингу.

Нейронні мережі здатні виявляти та класифікувати об'єкти навіть на зашумлених зображеннях, розрізняти подібні класи об'єктів і адаптуватися до різних умов спостереження. Однак їх успішне впровадження вимагає вирішення ряду наукових та технічних задач, пов'язаних з оптимізацією архітектур, навчанням моделей на репрезентативних наборах даних та інтеграцією з традиційними методами відстеження і прогнозування руху.

Метою цієї роботи є розробка та вдосконалення методів визначення, класифікації та прогнозування руху об'єктів у повітряному просторі на основі глибокого навчання, що забезпечують підвищену точність розпізнавання та надійний трекінг за різних умов спостереження. Особлива увага приділяється здатності системи ефективно розрізняти дрони та птахів, що є критичним аспектом для мінімізації помилкових спрацьовувань.

Результати дослідження можуть бути застосовані в різних сферах, включаючи забезпечення безпеки критичної інфраструктури, моніторинг повітряного простору над аеропортами та іншими об'єктами підвищеної уваги, контроль за дотриманням заборон на польоти БПЛА у певних зонах, а також для наукових досліджень поведінки птахів. Запропоновані методи можуть стати основою для створення комплексних систем виявлення несанкціонованих польотів та відстеження потенційно небезпечних об'єктів.

Варто зазначити, що задача автоматичного відстеження об'єктів у повітряному просторі має тривалу історію розвитку, але сучасні виклики вимагають принципово нових підходів. Поява нейромережевих алгоритмів комп'ютерного зору, зростання обчислювальних потужностей та розвиток сенсорних технологій створюють унікальну можливість для якісного стрибка в ефективності систем моніторингу. Водночас, практичне впровадження таких систем вимагає ретельного балансування між точністю, надійністю та вимогами до обчислювальних ресурсів, особливо в контексті роботи в режимі реального часу.

## **1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ТЕОРЕТИЧНІ ОСНОВИ ВІДСТЕЖЕННЯ ОБ'ЄКТІВ У ПОВІТРЯНОМУ ПРОСТОРИ**

1.1 Актуальність відстеження та прогнозування руху об'єктів у повітряному просторі

Моніторинг повітряного простору в сучасних умовах стикається з низкою викликів, які роблять традиційні підходи до спостереження недостатньо ефективними.

Повітряний простір сьогодні заповнений різноманітними об'єктами: від великих літаків до малих безпілотних літальних апаратів (БПЛА), птахів та інших природних об'єктів. Здатність розрізняти ці об'єкти є критично важливою для ефективного моніторингу.

За даними Міжнародної організації цивільної авіації (ІСАО), обсяг повітряних перевезень зростає приблизно на 5–6% щорічно [1], що призводить до збільшення кількості об'єктів, які потрібно одночасно відстежувати.

Традиційні радарні системи мають обмеження щодо виявлення малорозмірних об'єктів, особливо тих, що виготовлені з матеріалів з низькою відбивною здатністю, таких як сучасні БПЛА.

Сучасні системи моніторингу повинні обробляти величезні обсяги даних і приймати рішення в режимі реального часу, що вимагає високої обчислювальної потужності та ефективних алгоритмів.

Для ефективного управління повітряним рухом необхідно не лише відстежувати поточний стан об'єктів, а й прогнозувати їх майбутні траєкторії з високою точністю.

Особливе місце серед викликів моніторингу повітряного простору займає стрімке зростання використання БПЛА різного призначення. За даними аналітичних агентств, глобальний ринок комерційних дронів

зростає в середньому на 25–30% щорічно [2]. Прогнозується, що до 2028 року загальна кількість дронів у світі перевищить 30 мільйонів одиниць.

БПЛА можуть використовуватися для проникнення в захищені зони, включаючи військові об'єкти, атомні електростанції, урядові будівлі та інші об'єкти критичної інфраструктури.

Зіткнення дрона з пасажирським літаком може призвести до серйозних наслідків. За даними TSA (управління з безпеки транспорту), кількість повідомлень про небезпечне наближення дронів до літаків зростає в кілька разів за останні роки [3].

Обладнані вибуховими пристроями або хімічними речовинами дрони можуть становити серйозну загрозу для громадської безпеки. БПЛА з камерами можуть використовуватися для несанкціонованого стеження або збору інформації. Дрони можуть бути використані для отримання комерційної або промислової інформації.

Дані таблиці 1.1 свідчать про стійке зростання кількості інцидентів, пов'язаних з БПЛА, що підкреслює актуальність розробки систем моніторингу, здатних ефективно виявляти та відстежувати такі об'єкти.

Таблиця 1.1 – Динаміка зростання кількості зареєстрованих інцидентів з БПЛА в США (2017–2023 рр.)

Рік	Кількість зареєстрованих інцидентів	Приріст до попереднього року, %
2017	1,771	-
2018	2,308	30.3
2019	2,723	18.0
2020	2,187	-19.7
2021	2,629	20.2
2022	3,124	18.8
2023	3,749	20.0

Розвиток систем відстеження та прогнозування руху об'єктів у повітряному просторі має критичне значення для багатьох галузей. Особливо актуальними є рішення, засновані на сучасних технологіях комп'ютерного зору та глибокого навчання, які дозволяють підвищити точність виявлення, класифікації та прогнозування траєкторій руху різноманітних об'єктів.

Таким чином, розробка ефективних методів визначення, класифікації та прогнозування рухомих об'єктів у повітряному просторі на основі глибокого навчання є актуальним науково-технічним завданням, що має значний практичний потенціал застосування в різних сферах діяльності.

## 1.2 Аналіз предметної галузі

Системи моніторингу повітряного простору можна класифікувати за різними критеріями, що визначають їх функціональні можливості, технологічну базу та сфери застосування.

За принципом дії та використовуваними технологіями системи моніторингу діляться на:

- радіолокаційні системи – базуються на принципі активної локації, випромінюючи радіосигнали та аналізуючи їх відбиття від об'єктів;
- оптико-електронні системи – використовують візуальний спектр та інфрачервоне випромінювання;
- акустичні системи – базуються на виявленні та аналізі звукових хвиль;
- багатосенсорні системи – комбінують різні типи датчиків для підвищення точності та надійності;
- локальні системи – призначені для захисту конкретних об'єктів.

За рівнем автоматизації та використання штучного інтелекту системи моніторингу діляться на:

- традиційні системи – з переважно ручним аналізом даних;

- напіваавтоматичні системи – автоматизують окремі функції при збереженні контролю оператора;
- високоавтоматизовані системи – використовують алгоритми машинного навчання і штучного інтелекту для автономної роботи.

Це порівняння систем моніторингу можна побачити у таблиці 1.2.

Таблиця 1.2 – Порівняння основних типів систем моніторингу повітряного простору

Тип системи	Переваги	Обмеження	Типові сфери застосування
Первинні радари	Незалежність від обладнання на об'єкті, всепогодність	Висока вартість, обмежена роздільна здатність	Управління повітряним рухом, військове спостереження
Вторинні радари	Отримання додаткової інформації про об'єкт, висока точність	Залежність від транспондера на об'єкті	Цивільне управління повітряним рухом
Оптико-електронні системи	Висока роздільна здатність, візуальна ідентифікація	Залежність від погодних умов, обмежена дальність	Захист об'єктів, виявлення малих БПЛА
Акустичні системи	Низька вартість, виявлення неспостережуваних за іншими каналами об'єктів	Залежність від зовнішніх шумів, обмежена дальність	Виявлення малих БПЛА в міських умовах
Багатосенсорні системи	Комплексне виявлення, висока точність	Складність інтеграції, висока вартість	Критично важливі об'єкти

Повітряний простір населяють різноманітні об'єкти, які відрізняються за своїми характеристиками, що впливає на методи їх виявлення, класифікації та відстеження.

Пілотовані літальні апарати:

- пасажирські літаки (велика ЕПР, висока швидкість, передбачувані траєкторії);
- військові літаки (змінна ЕПР, висока швидкість, складні маневри);
- вертольоти (середня ЕПР, низька/середня швидкість, унікальні акустичні сигнатури);
- легкі літаки та аеростати (мала/середня ЕПР, низька швидкість).

Безпілотні літальні апарати (БПЛА):

- військові БПЛА (середня ЕПР, висока автономність, складні алгоритми польоту);
- комерційні дрони (мала ЕПР, переважно квадрокоптери, обмежена автономність);
- споживацькі дрони (дуже мала ЕПР, швидкість до 70 км/год, обмежений час польоту);
- спеціалізовані БПЛА (різноманітні характеристики залежно від призначення).

Інші штучні об'єкти:

- ракети та снаряди (мала ЕПР, дуже висока швидкість, балістичні траєкторії);
- супутники та космічні апарати (в межах видимості);
- метеозонди та повітряні кулі (мала ЕПР, залежність від повітряних потоків).

Птахи:

- великі (орли, лелеки) – ЕПР близька до малих дронів, швидкість до 80 км/год;

- середні (голуби, ворони) – мала ЕПР, швидкість до 60 км/год;
- дрібні та зграї – дуже мала індивідуальна ЕПР, можуть формувати значні відбиття у зграї.

Метеорологічні явища:

- хмари;
- опади;
- турбулентності та повітряні потоки.

ЕПР – ефективна площа розсіювання, характеристика помітності об'єкта для радіолокаційних систем.

Особливу складність для систем моніторингу представляє розрізнення птахів та малих дронів, оскільки ці об'єкти мають подібні габарити, швидкості руху та можуть демонструвати схожі патерни польоту. Як видно з рисунка 1.1, зона характеристик цих об'єктів суттєво перетинається, що вимагає застосування додаткових критеріїв класифікації [4].

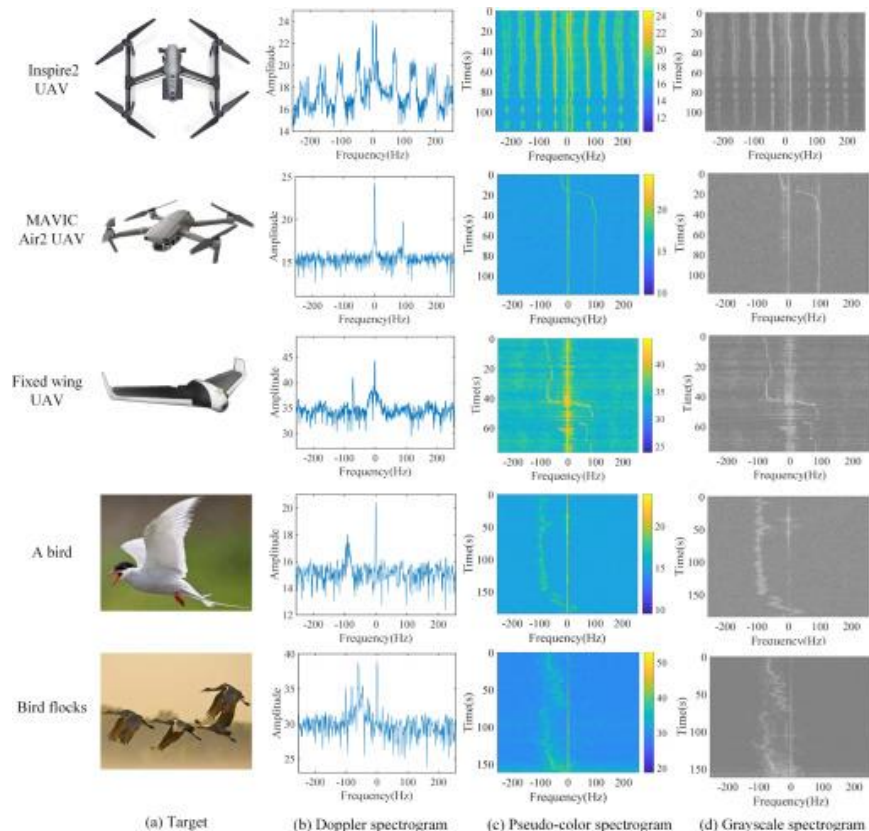


Рисунок 1.1 – Порівняння характеристик БПЛА та птахів

### 1.3. Технологічні аспекти побудови систем відстеження

Створення ефективної системи відстеження рухомих об'єктів у повітряному просторі вимагає вирішення комплексу технологічних завдань, що охоплюють повний цикл обробки інформації. Архітектура таких систем зазвичай включає п'ять взаємозалежних рівнів.

Сенсорний рівень забезпечує збір первинних даних через різноманітні датчики: радіолокаційні станції, оптичні та інфрачервоні камери, мікрофонні решітки, лідари та датчики радіовипромінювання. Ці компоненти формують первинну інформаційну базу для аналізу.

На рівні первинної обробки відбувається перетворення сигналів датчиків у придатні для аналізу формати через фільтрацію шумів, виділення потенційних об'єктів та параметризацію сигналів. Цей етап є критичним для забезпечення якості вхідних даних для наступних рівнів.

Рівень детекції та класифікації здійснює аналіз оброблених даних за допомогою алгоритмів комп'ютерного зору, методів машинного навчання та глибоких нейронних мереж, а також технік спектрального та часового аналізу. Тут відбувається ідентифікація об'єктів та їх віднесення до відповідних класів [5].

Рівень відстеження та прогнозування забезпечує побудову і підтримку моделей руху об'єктів через алгоритми асоціації даних, фільтри для згладжування траєкторій та моделі прогнозування руху, що дозволяє прогнозувати майбутні положення цілей.

Завершальний рівень прийняття рішень інтерпретує отримані результати та формує рекомендації через системи правил для класифікації загроз, інтерфейси оператора та механізми автоматичного реагування на виявлені об'єкти та ситуації.

Ефективна інтеграція різних рівнів системи створює надійну платформу моніторингу повітряного простору, здатну виявляти, класифікувати та прогнозувати рух різноманітних об'єктів. Серед ключових

технологічних викликів при розробці таких систем варто відзначити необхідність об'єднання інформації з різних типів датчиків для підвищення точності та надійності виявлення. Також критичною задачею є забезпечення аналізу відеопотоків високої роздільної здатності та обробка багатоканальних даних з мінімальною затримкою. Розробникам необхідно постійно балансувати між точністю та обчислювальною ефективністю, оптимізуючи алгоритми для роботи на доступних апаратних платформах без компромісів щодо якості. Важливою вимогою є стійкість до завад та спроб маскування, що забезпечує надійну роботу в умовах навмисних перешкод. Не менш важливим аспектом залишається захист від кіберзагроз через запобігання несанкціонованому доступу або маніпуляціям з даними систем моніторингу.

Апаратна платформа систем відстеження може варіюватися від централізованих серверних рішень до розподілених систем з елементами граничних обчислень (edge computing):

- централізовані системи – всі дані передаються в центр обробки для аналізу, що забезпечує доступ до значних обчислювальних ресурсів, але створює вимоги до каналів зв'язку та може збільшувати затримки;
- розподілені системи – частина обробки виконується безпосередньо на сенсорних вузлах або локальних обчислювальних пристроях, що дозволяє зменшити обсяг даних, які передаються, та скоротити затримки;
- гібридні системи – адаптивно розподіляють обчислення між локальними вузлами та центральними серверами залежно від навантаження та доступних ресурсів.

Порівняння ключових компонентів систем відстеження можемо побачити на таблиці 1.3 [6].

Таблиця 1.3 – Технологічна зрілість ключових компонентів систем відстеження

Технологія	Рівень зрілості	Перспективні напрямки розвитку
Комп'ютерний зір	Високий для стандартних умов, середній для складних умов	Всепогодне розпізнавання, робота при частковому затіненні
Акустичне виявлення	Середній	Інтелектуальна фільтрація шумів, розпізнавання за акустичною сигнатурою
Радіолокаційне виявлення малих об'єктів	Середній	Підвищення роздільної здатності, зниження мінімальної ЕПР
Багатосенсорна інтеграція	Середній/Низький	Адаптивне зважування інформації з різних джерел
Розпізнавання намірів	Низький	Аналіз патернів поведінки, передбачення маневрів

Аналіз предметної галузі демонструє, що ефективне відстеження та прогнозування руху об'єктів у повітряному просторі вимагає комплексного підходу, який поєднує передові методи обробки сигналів, комп'ютерного зору, глибокого навчання та інженерії знань. Особливу увагу слід приділити розрізненню невеликих БПЛА та птахів, що залишається складним завданням для існуючих систем.

#### 1.4 Теоретичні основи визначення та класифікації рухомих об'єктів

Розвиток методів детекції та класифікації об'єктів пройшов кілька ключових етапів, кожен з яких характеризувався принципово відмінними

підходами та рівнем ефективності. Цей розвиток можемо побачити у таблиці 1.4.

Таблиця 1.4 – Порівняння ключових методів детекції об'єктів

Метод	Тип	Рік	mAP (COCO)	FPS (GPU)	Особливості
Viola-Jones	Традиційний	2001	-	~5	Каскадний підхід, обмежений набір об'єктів
HOG+SVM	Традиційний	2005	-	~1	Висока чутливість до варіацій об'єктів
Faster R-CNN	Двостадійний	2015	34.9%	12	Висока точність, помірна швидкість
YOLOv3	Одностадійний	2018	33.0%	45	Баланс швидкості та точності
DETR	Трансформер-базований	2020	52.9%	155	Інноваційна архітектура без анкорів
YOLOv8	Одностадійний	2023	42.0%	28	Сучасний стан техніки

На першому етапі домінували методи, засновані на ручному проєктуванні ознак:

- Віола-Джонс (2001) – каскадний класифікатор з використанням ознак Хаара, що забезпечував швидку детекцію обличчя за рахунок інтегрального представлення зображення [7];
- гістограми орієнтованих градієнтів (HOG, 2005) – метод, що зосереджується на розподілі градієнтів інтенсивності для характеристики локальної форми об'єкта;

- Scale-Invariant Feature Transform (SIFT, 1999) – алгоритм для виявлення та опису локальних ознак на зображеннях, інваріантний до масштабування та обертання;

- Optical Flow – методи для оцінки руху об'єктів між послідовними кадрами відеопотоку.

Ці методи мали обмежену здатність до узагальнення та потребували значного налаштування для кожної конкретної задачі.

На другому етапі, з появою та розвитком глибоких нейронних мереж, відбувся перший якісний стрибок у точності детекції:

- R-CNN (Regions with CNN features, 2014) – перший успішний підхід, що інтегрував згорткові нейронні мережі з методами виділення регіонів-кандидатів;

- Fast R-CNN (2015) – оптимізована версія R-CNN, що значно прискорила процес детекції за рахунок спільних обчислень для всього зображення;

- Faster R-CNN (2015) – впровадження Region Proposal Network (RPN) для генерації регіонів-кандидатів безпосередньо в мережі, створення першої end-to-end системи детекції.

Третій етап характеризується розвитком ефективних архітектур та алгоритмів, що забезпечують високу точність при реальночасовій обробці:

- YOLO (You Only Look Once, 2016) – піонер одностадійних детекторів, що розглядає детекцію як задачу регресії, забезпечуючи високу швидкість обробки [8];

- SSD (Single Shot MultiBox Detector, 2016) – використовує багатомасштабну мережу зі спеціальними шарами для виявлення об'єктів різного розміру;

- RetinaNet (2017) – впровадження Focal Loss для вирішення проблеми дисбалансу класів;

- EfficientDet (2020) – архітектура, що оптимізує баланс між точністю та обчислювальною ефективністю;

– YOLOv4-v8 (2020–2023) – сучасні версії YOLO з численними архітектурними вдосконаленнями та налаштуваннями [9].

У відстеженні об'єктів на відео також спостерігався значний прогрес: традиційні трекери (Mean-Shift, Kalman Filter), трекери з глибоким навчанням (SORT, DeepSORT, ByteTrack) та інтегровані детектори-трекери (YOLOX) [10].

Поряд із розвитком методів детекції вдосконалювались і підходи до класифікації типів рухомих об'єктів:

- статичні класифікатори – розглядають кожен кадр окремо;
- рекурентні архітектури (RNN, LSTM) – враховують темпоральну інформацію;
- 3D-згорткові мережі – одночасна обробка просторової та часової інформації;
- Transformer-based моделі – сучасний підхід з механізмом уваги для інтеграції інформації з різних кадрів.

На рисунку 1.2 ми можемо побачити цей розвиток.

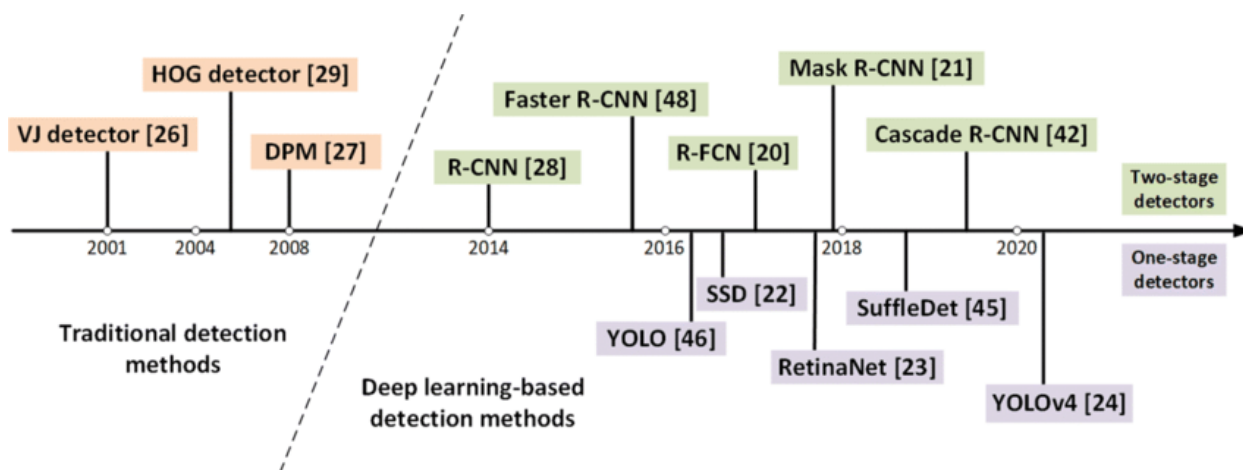


Рисунок 1.2 – Еволюція архітектур нейронних мереж для детекції об'єктів

Детекція та класифікація об'єктів у повітряному просторі має ряд специфічних особливостей, що відрізняють її від стандартних завдань комп'ютерного зору [11].

Специфіка об'єктів спостереження полягає насамперед у їхніх малих розмірах – дрони та птахи можуть займати лише декілька пікселів на зображенні, що значно ускладнює їх виявлення та класифікацію. Додаткову складність створює висока швидкість руху об'єктів, які швидко переміщуються через поле зору камери, викликаючи розмиття та спотворення зображення. Варіативність форм та розмірів становить ще одну проблему, адже навіть у межах однієї категорії (наприклад, дрони) об'єкти можуть значно відрізнятися за зовнішнім виглядом. Складна динаміка руху також ускладнює завдання – об'єкти в повітрі мають шість ступенів свободи, їх рух може бути непередбачуваним та сильно варіюватися.

Особливості середовища спостереження створюють додаткові виклики. На відміну від наземного спостереження, фоном слугує динамічне небо з хмарами, змінами освітлення та атмосферними явищами, що ускладнює виділення об'єктів. Погодні умови, такі як дощ, туман, сніг, значно впливають на видимість і можуть повністю блокувати виявлення оптичними засобами. Необмежений простір спостереження означає, що об'єкт може з'явитися в будь-якій точці тривимірного простору, що значно розширює область пошуку. Зміна освітлення через сонячне засліплення, тіні від хмар, зміну часу доби створює змінні умови для виявлення.

Вимоги до систем детекції включають обробку в реальному часі, оскільки затримка у виявленні може призвести до втрати відстеження об'єкта. Мінімізація хибних спрацьовувань критично важлива, особливо в системах безпеки, де помилкові тривоги знижують довіру до системи. Висока чутливість необхідна для виявлення малорозмірних об'єктів на великих відстанях, а стійкість до атмосферних явищ забезпечує роботу в різних погодних умовах.

Для вирішення цих завдань використовуються спеціалізовані архітектури нейронних мереж – модифікації стандартних архітектур для роботи з малими об'єктами. Вони включають додаткові шляхи виявлення для малих об'єктів, багатомасштабне об'єднання ознак та механізми уваги для виділення релевантних областей. Методи попередньої та післяобробки, такі як адаптивна фільтрація шумів, покращення контрастності та часова інтеграція виявлень, підвищують якість детекції. Підходи до навчання включають несиметричні функції втрат для роботи з незбалансованими класами, специфічну аугментацію даних для повітряних об'єктів та зважування прикладів за складністю.

Одним із найскладніших завдань у цій галузі є надійне розрізнення птахів та малих дронів, оскільки вони мають близькі характеристики руху та розміри. Це порівняння ми можемо побачити у таблиці 1.5 [12].

Таблиця 1.5 – Порівняння характеристик детекції різних типів повітряних об'єктів

Тип об'єкта	Основні виклики	Найбільш ефективні підходи	Типова точність (mAP)
Великі літаки	Висока швидкість, значні зміни ракурсу	Стандартні детектори з адаптацією	85–95%
Малі БПЛА	Малий розмір, варіативність вигляду	Спеціалізовані архітектури з виділенням малих об'єктів	65–80%
Птахи	Малий розмір, органічна форма, складна динаміка	Аналіз послідовності кадрів, трекінг	60–75%
Зграї птахів	Множинні взаємодіючі об'єкти	Щільніший аналіз, колективна динаміка	70–85%

Для вирішення задачі розрізнення птахів та малих дронів використовується аналіз патернів руху (птахи мають більш органічні, плавні рухи, тоді як дрони демонструють механічні, геометрично правильні патерни), частотний аналіз траєкторії, вивчення морфологічних ознак та застосування темпоральних згорткових мереж для аналізу послідовностей кадрів.

Теоретичні основи визначення та класифікації рухомих об'єктів у повітряному просторі охоплюють широкий спектр підходів, від класичних методів комп'ютерного зору до сучасних архітектур глибокого навчання [13]. Ключовою особливістю є необхідність адаптації загальних методів до специфічних умов повітряного спостереження та вирішення проблеми розрізнення об'єктів з подібними характеристиками, таких як дрони та птахи. Ефективні системи повинні поєднувати передові методи детекції з алгоритмами відстеження для забезпечення стабільного функціонування в різноманітних умовах експлуатації.

### 1.5 Постановка завдання дослідження

Основною метою дослідження є розробка та вдосконалення методів визначення, класифікації та прогнозування руху об'єктів у повітряному просторі на основі глибокого навчання, що забезпечують підвищену точність розпізнавання та надійний трекінг в умовах обмеженої видимості, зі здатністю ефективно розрізняти дрони та птахів.

Для досягнення поставленої мети в роботі необхідно вирішити комплекс взаємопов'язаних завдань. Першочерговим завданням є проведення ґрунтового аналізу та порівняння існуючих методів детекції об'єктів у повітряному просторі. Це включає вивчення сучасних архітектур нейронних мереж, придатних для виявлення малорозмірних об'єктів, дослідження особливостей їх застосування для специфічної задачі

розрізнення дронів та птахів, а також визначення ключових обмежень існуючих підходів.

На основі проведеного аналізу необхідно розробити метод підвищення точності класифікації повітряних об'єктів. Це потребує створення та оптимізації архітектури нейронної мережі, спеціалізованої для класифікації дронів, птахів та інших об'єктів, що можуть перебувати у повітряному просторі. Важливим аспектом є розробка підходу до формування навчальної вибірки з урахуванням специфіки завдання, а також впровадження механізмів, що забезпечують стійкість системи до змін умов спостереження, таких як варіації освітлення та погодні умови.

Паралельно необхідно удосконалити методи трекінгу об'єктів на основі візуальної інформації. Це завдання передбачає розробку алгоритму відстеження, який враховує особливості руху повітряних цілей, забезпечує стабільне відстеження навіть в умовах тимчасової втрати об'єкта через перешкоди та інтегрує механізми фільтрації даних для згладжування траєкторій.

Критично важливим завданням є створення методу прогнозування траєкторій руху на основі комбінації традиційних алгоритмів та глибокого навчання. Розроблена модель прогнозування повинна враховувати фізичні характеристики різних типів об'єктів, мати механізми адаптації до змін режиму руху об'єкта та забезпечувати оцінку невизначеності прогнозу, що особливо важливо для практичного застосування.

Інтеграція всіх розроблених методів у єдину систему є наступним важливим завданням. Необхідно створити архітектуру, яка ефективно об'єднує функції детекції, класифікації та прогнозування, забезпечує роботу в режимі реального часу на доступному обчислювальному обладнанні та має інтерфейси для обміну даними з іншими системами.

Завершальним етапом є експериментальна оцінка розроблених методів. Для цього необхідно створити репрезентативний набір тестових даних, що відображає реальні умови експлуатації, провести порівняльні

дослідження з існуючими аналогами та оцінити ефективність системи за визначеними критеріями продуктивності.

Вирішення цих взаємопов'язаних завдань дозволить створити ефективну систему для виявлення, класифікації та прогнозування руху дронів у повітряному просторі, здатну працювати в різноманітних умовах та забезпечувати високу точність результатів.

При розробці методів визначення, класифікації та прогнозування руху об'єктів у повітряному просторі необхідно враховувати ряд обмежень та робити певні припущення, які впливають на проектування та практичну реалізацію системи.

В контексті обмежень вхідних даних слід зазначити, що система орієнтована на роботу з об'єктами розміром не менше  $5 \times 5$  пікселів на зображенні, оскільки менші розміри не надають достатньо інформації для надійної класифікації. Система розрахована на обробку відеопотоку з частотою не менше 15 кадрів на секунду, що забезпечує достатню темпоральну інформацію для ефективного трекінгу та прогнозування. Щодо метеорологічних умов, передбачається функціонування системи при видимості не менше 1 км та за відсутності сильних опадів, які можуть суттєво погіршувати якість відеоданих.

Обмеження обчислювальних ресурсів є ключовими для практичного впровадження системи. Передбачається обробка в режимі реального часу на обладнанні з продуктивністю не нижче NVIDIA GeForce GTX 1080 Ti, використання не більше 4 ГБ оперативної пам'яті для роботи системи та забезпечення затримки між отриманням кадру та формуванням результату не більше 100 мс, що є критичним для систем реального часу.

Функціональні обмеження системи включають можливість одночасного відстеження не більше 20 об'єктів, що є достатнім для більшості практичних сценаріїв. Система орієнтована на роботу з об'єктами, що рухаються зі швидкістю до 150 км/год для дронів та до 100 км/год для птахів, що охоплює переважну більшість малих БПЛА та птахів. Горизонт

прогнозування обмежений 10 секундами для об'єктів з регулярним рухом, що є достатнім для прийняття тактичних рішень.

Щодо характеру руху об'єктів, робляться припущення, що дрони переважно здійснюють цілеспрямований рух з обмеженою кількістю різких маневрів, птахи демонструють більш плавні та органічні рухи з природними коливаннями, а об'єкти загалом тяжіють до збереження режиму руху протягом короткого періоду часу. Ці припущення відповідають спостереженням за реальною поведінкою відповідних об'єктів та дозволяють більш ефективно налаштувати алгоритми трекінгу.

Для середовища спостереження припускається наявність статичної камери або камери з відомими параметрами руху, що спрощує обробку відеоданих, достатній контраст між об'єктами та фоном для більшості випадків, а також відсутність навмисного маскування об'єктів. Ці припущення відповідають типовим сценаріям моніторингу повітряного простору.

Щодо функціонування системи, передбачається наявність попередньо навченої моделі для початкової детекції об'єктів, можливість періодичного донавчання системи на нових даних та наявність механізмів калібрування та налаштування параметрів. Ці припущення забезпечують гнучкість та адаптивність системи до змін умов експлуатації.

Зазначені обмеження та припущення створюють чіткі рамки для розробки системи, дозволяючи зосередитись на найбільш практично значущих аспектах та забезпечити реалістичність її впровадження.

## 2 МЕТОДИ ВИЗНАЧЕННЯ, КЛАСИФІКАЦІЇ ТА ПРОГНОЗУВАННЯ РУХУ ДРОНІВ

### 2.1 Обґрунтування методів дослідження

Для ефективного вирішення задачі виявлення та відстеження дронів доцільно використовувати комплексний підхід, що враховує специфіку повітряних об'єктів. Ключовою вимогою до такої системи є забезпечення балансу між точністю виявлення та швидкістю обробки, що дозволить функціонувати в режимі реального часу.

Порівняльний аналіз нейромережових архітектур показує, що одностадійні детектори, зокрема YOLO (You Only Look Once), демонструють оптимальне співвідношення швидкодії та точності для виявлення малорозмірних об'єктів у повітряному просторі [14]. Серед різних версій цієї архітектури YOLOv8 забезпечує найкращі показники при збереженні високої швидкості обробки на стандартному обладнанні.

Варто впровадити методику підготовки збалансованого навчального набору даних, що включатиме, адаптивне вилучення кадрів з відеопотоків, комбінування зображень дронів і птахів різних типів, аугментацію даних для моделювання різних умов освітлення та погоди та збалансування класів для підвищення точності класифікації.

Такий методологічний підхід слід доповнити конвеєром обробки відеоданих з нормалізацією розміру, корекцією кольору та адаптивним контрастуванням, що підвищить якість вхідних даних і, відповідно, точність роботи детектора.

Поєднання зазначених методів дозволить створити систему, яка ефективно виявлятиме та відстежуватиме дрони, надійно розрізняючи їх від птахів навіть у складних умовах спостереження.

З формальної точки зору, задачу детекції об'єктів можна визначити як:

- вхідні дані: зображення і або кадр відеопотоку;

- вихідні дані: набір обмежувальних прямокутників (bounding boxes)  $b = \{b_1, b_2, \dots, b_n\}$  та відповідних їм класів  $c = \{c_1, c_2, \dots, c_n\}$ ;
- мета: мінімізація помилок локалізації та класифікації.

Для кількісної оцінки якості детекції використовуються різноманітні метрики.

Precision (точність) – частка правильних передбачень серед усіх передбачень, визначається формулою:

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}). \quad (2.1)$$

Recall (повнота) – частка виявлених об'єктів серед усіх наявних, визначається за формулою:

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}). \quad (2.2)$$

F1-score – гармонічне середнє між точністю та повнотою, визначається за формулою:

$$\text{F1} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall}). \quad (2.3)$$

Intersection over Union (IoU) – метрика якості локалізації, що обчислюється як відношення площі перетину передбаченого та реального обмежувальних прямокутників до площі їх об'єднання, можна вирахувати за формулою:

$$\text{IoU} = (\text{Area of Overlap}) / (\text{Area of Union}). \quad (2.4)$$

mAP (mean Average Precision) – середня точність при різних порогах IoU та для різних класів.

Рівень складності задачі детекції значно варіюється залежно від низки взаємопов'язаних факторів. Суттєвий вплив має кількість класів об'єктів, які система повинна розрізняти, а також варіативність зовнішнього вигляду представників одного класу [15]. Важливу роль відіграють розміри об'єктів відносно загального зображення, особливо коли йдеться про виявлення малорозмірних цілей. Умови освітлення та контрастність сцени можуть критично вплинути на якість детекції, так само як і наявність перекриттів та оклюзій, що частково приховують об'єкти. Швидкість руху цілей створює додаткові складнощі через можливе розмиття, а якість вхідних даних, включаючи роздільну здатність та рівень шумів, безпосередньо визначає базові можливості системи. У контексті відеопослідовностей задача ускладнюється необхідністю враховувати темпоральну інформацію – зміни стану об'єктів між кадрами. Це додає додатковий вимір складності, але водночас надає цінну інформацію про рух та поведінку об'єктів.

Практична реалізація системи виявлення та відстеження дронів потребує комплексного підходу до розробки програмної архітектури, оптимізації обчислювальних процесів та забезпечення інтеграції з різними джерелами даних.

Основою системи має бути модульна структура, побудована за об'єктно-орієнтованою парадигмою, що забезпечує необхідну гнучкість і масштабованість. Архітектура має формуватись навколо чотирьох ключових функціональних модулів: обробки даних, детекції об'єктів, прогнозування траєкторій та візуалізації результатів. Взаємодія між компонентами буде організована за допомогою патерну «спостерігач», що дозволяє модулям обмінюватися інформацією без зайвого зв'язування та залежностей.

Потік даних у такій системі функціонує як конвеєр, де вхідне відео проходить поетапну обробку: від зчитування і попередньої обробки кадрів до виявлення об'єктів, їх асоціації з існуючими треками, прогнозування траєкторій та формування візуального представлення результатів. Для

ефективного обміну даними між паралельними процесами застосовується механізм черг, що забезпечує безпечний доступ до спільних ресурсів.

Для забезпечення роботи в режимі реального часу має бути впроваджено комплекс оптимізацій на різних рівнях системи. Паралельна обробка даних дозволить розподілити обчислювальні задачі між кількома процесами, що суттєво підвищує продуктивність.

Важливим технологічним аспектом є ефективне управління пам'яттю. Може застосовуватись повторне використання буферів, оптимізоване копіювання даних між CPU та GPU, а також контроль життєвого циклу ресурсомістких об'єктів.

Архітектурні рішення системи ґрунтуються на принципах SOLID та використанні відповідних шаблонів проектування. Інверсія залежностей забезпечує гнучкість у заміні компонентів, фабричні методи спрощують вибір реалізацій на етапі виконання, а перевага композиції над наслідуванням дозволяє ефективно комбінувати функціональність.

## 2.2 Методи виявлення та класифікації повітряних об'єктів

Виявлення та класифікація повітряних об'єктів становлять фундаментальну частину розробленої системи відстеження дронів. У контексті нашого проекту ці методи мають забезпечувати надійне розпізнавання малорозмірних об'єктів, зокрема безпілотних літальних апаратів, та їх розрізнення від інших об'єктів, таких як птахи.

Основним компонентом системи виявлення є нейронна мережа YOLOv8, яку ми адаптували для специфічних потреб відстеження дронів. Архітектура YOLOv8 (You Only Look Once, version 8) належить до класу одностадійних детекторів, які здійснюють виявлення та класифікацію об'єктів за один прохід через мережу, що забезпечує високу швидкодію при збереженні прийнятної точності.

Принцип роботи YOLO базується на розділенні вхідного зображення на сітку розміром  $S \times S$ . Для кожної комірки сітки мережа передбачає  $B$  обмежувальних прямокутників (bounding boxes), для кожного з яких розраховуються: координати центру ( $x, y$ ), ширина і висота ( $w, h$ ), достовірність виявлення (confidence) та ймовірності приналежності до різних класів. Математично це можна представити як представлено на формулі:

$$P(\text{Object}) \times \text{IOU}(\text{pred}, \text{truth}) = \text{Confidence}, \quad (2.5)$$

де  $P(\text{Object})$  – ймовірність наявності об'єкта в комірці;

IOU (Intersection over Union) – міра перекриття між передбаченим та справжнім обмежувальними прямокутниками.

Функція втрат у YOLOv8 композитна і складається з трьох компонентів: втрати локалізації (для координат та розмірів), втрати достовірності (для оцінки наявності об'єкта) та втрати класифікації (для визначення класу об'єкта) [16], як показано на формулі:

$$L = \lambda_{\text{coord}} \times L_{\text{loc}} + \lambda_{\text{conf}} \times L_{\text{conf}} + \lambda_{\text{class}} \times L_{\text{class}}, \quad (2.6)$$

де  $\lambda_{\text{coord}}$ ,  $\lambda_{\text{conf}}$ ,  $\lambda_{\text{class}}$  – вагові коефіцієнти для балансування різних компонентів втрат.

Для підвищення ефективності розпізнавання малорозмірних об'єктів, якими часто є дрони та птахи у відеопотоці, ми модифікували стандартну архітектуру YOLOv8. Зокрема, були впроваджені такі адаптації:

Розширення шляхів ознак (feature paths) для малих об'єктів шляхом додавання додаткових з'єднань у FPN (Feature Pyramid Network), що дозволяє краще зберігати дрібні деталі.

Впровадження модуля просторово-канальної уваги (Spatial-Channel Attention Module), який дозволяє мережі фокусуватися на найбільш

інформативних регіонах та каналах даних, що особливо важливо для виявлення малокоонтрастних об'єктів на однорідному фоні неба.

Механізми уваги довели свою ефективність для задач детекції у різних спектрах [17].

Focal Loss дозволяє ефективно боротися з дисбалансом класів, що особливо важливо при детекції малорозмірних об'єктів на складному фоні [18]. Заміна стандартної функції втрат для класифікації на Focal Loss, визначається формулою:

$$FL(pt) = -\alpha t \times (1-pt)^\gamma \times \log(pt), \quad (2.7)$$

де  $pt$  – ймовірність правильного класу;

$\alpha t$  – коефіцієнт для балансування класів,  $\gamma$  параметр фокусування (зазвичай  $\gamma = 2$ ).

Focal Loss надає більшої ваги складним для класифікації прикладам, що покращує розрізнення схожих класів об'єктів, таких як дрони та птахи.

### 2.3 Методи трекінгу та прогнозування траєкторії

Після успішного виявлення та класифікації об'єктів на окремих кадрах необхідно забезпечити їх стабільне відстеження та прогнозування траєкторій руху. Ключовим елементом розробленого нами підходу є фільтр Калмана, доповнений спеціалізованими алгоритмами асоціації даних та методами оцінки невизначеності.

Математичне обґрунтування вибору фільтра Калмана базується на його оптимальності для лінійних систем з гауссівським шумом, а також можливості ефективної адаптації до нелінійних систем через розширені модифікації [19]. У нашій реалізації використовується модель з постійною швидкістю для відстеження дронів. Вектор стану представлений формулою:

$$x(t) = [x, y, z, vx, vy, vz]^T, \quad (2.8)$$

де  $x, y, z$  – координати положення об'єкта у тривимірному просторі;

$vx, vy, vz$  – компоненти швидкості вздовж відповідних осей координат.

Рівняння стану має вигляд, як на формулах:

$$x(k+1) = F(k) * x(k) + w(k), \quad (2.9)$$

$$F(k) = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.10)$$

де  $\Delta t$  – часовий крок;

$w(k)$  – випадковий шум процесу, який моделюється як білий гауссівський шум з коваріаційною матрицею  $Q$ .

Матриця коваріації шуму процесу  $Q$  була спеціально налаштована для характеристик руху дронів з використанням блочної структури. Для нашої задачі емпірично визначено оптимальне значення стандартного відхилення прискорення  $\sigma_a = 0.1$ . Для вимірювань використовується модель:

$$z(k) = H(k)*x(k)+v(k), \quad (2.11)$$

де  $H(k)$  – матриця спостереження;

$v(k)$  – шум вимірювань з коваріаційною матрицею.

Фільтр Калмана це рекурсивний алгоритм оптимальної фільтрації, який об'єднує модель руху об'єкта з вимірюваннями для отримання найкращої оцінки його стану. Фільтр Калмана працює у два етапи.

Етап прогнозування, видно на формулах:

$$x_{k|k-1} = F_k x_{k-1|k-1}, \quad (2.12)$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k. \quad (2.13)$$

Етап коригування, можемо побачити на формулах:

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1}, \quad (2.14)$$

$$x_{k|k} = x_{k|k-1} + K_k (z_k - H_k x_{k|k-1}), \quad (2.15)$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}, \quad (2.16)$$

де  $x_{k|k-1}$  – прогнозований стан;

$P_{k|k-1}$  – коваріаційна матриця помилки прогнозу;

$K_k$  – коефіцієнт підсилення Калмана;

$z_k$  – вимірювання;

$F_k$  – матриця переходу станів;

$H_k$  – матриця спостереження;

$Q_k$  – коваріаційна матриця шуму процесу;

$R_k$  – коваріаційна матриця шуму вимірювань.

Також існують інші види фільтра Калмана, ось декілька прикладів:

- розширений фільтр Калмана (EKF) – для нелінійних систем;
- безрозмірний фільтр Калмана (UKF) – використовує сигма-точки замість лінеаризації;
- частковий фільтр – метод для сильно нелінійних систем;
- адаптивний фільтр Калмана.

Важливою перевагою фільтра Калмана є його здатність працювати з пропущеними вимірюваннями, що критично важливо для відстеження дронів у реальних сценаріях, коли об'єкт може тимчасово зникати з поля зору або бути перекритим.

Математичне моделювання траєкторій руху є фундаментальною основою для систем прогнозування [20]. Різні моделі забезпечують різний баланс між точністю, обчислювальною складністю та інтерпретованістю [21]. Кінематичні моделі, які ми використовуємо, описують рух об'єкта без урахування фізичних сил.

Крім базової моделі постійної швидкості (CV), може використовуватись модель постійного прискорення (CA), яка розширює вектор стану:

$$x(t) = [x, y, z, vx, vy, vz, ax, ay, az]T, \quad (2.17)$$

де  $x, y, z$  – координати положення об'єкта;

$vx, vy, vz$  – компоненти швидкості;

$ax, ay, az$  – компоненти прискорення.

Ця модель краще підходить для об'єктів, що здійснюють маневри.

Траєкторію руху можна розглядати як багатовимірний часовий ряд – послідовність спостережень, упорядкованих у часі. Часовий ряд траєкторії може бути представлений у різних форматах, наприклад, як послідовність координат положення.

Послідовність координат положення можемо побачити на формулі:

$$T = \{(x_1, y_1, z_1, t_1), (x_2, y_2, z_2, t_2), \dots, (x_n, y_n, z_n, t_n)\}. \quad (2.18)$$

Розширений вектор стану можемо побачити на формулі:

$$T = \{(x_1, y_1, z_1, vx_1, vy_1, vz_1, t_1), \dots, (x_n, y_n, z_n, vx_n, vy_n, vz_n, t_n)\}. \quad (2.19)$$

Наприклад, для об'єкта, що рухається в горизонтальній площині, можна використовувати полярні координати, як показано на формулі:

$$T=\{(r_1,\theta_1,t_1),(r_2,\theta_2,t_2),\dots,(r_n,\theta_n,t_n)\}. \quad (2.20)$$

Фізичні моделі враховують закони фізики та аеродинаміки для більш точного опису.

Аеродинамічні моделі враховують підйомну силу, опір повітря та гравітацію.

Балістичні моделі використовуються для об'єктів, рух яких визначається гравітацією та початковими умовами.

Моделі поведінки застосовуються для об'єктів, керованих інтелектуальними агентами. Моделі намірів намагаються передбачити цілі об'єкта. Моделі на основі правил описують типові шаблони поведінки. Моделі колективної поведінки для опису руху груп об'єктів [22].

Вибір моделі залежить від типу об'єкта, характеру його руху та вимог до точності прогнозування.

Для забезпечення стабільного відстеження об'єктів ми впровадили спеціалізовані алгоритми асоціації даних, які дозволяють правильно зіставляти виявлені об'єкти з існуючими треками. Основою системи асоціації є алгоритм, що комбінує просторову близькість та візуальну схожість об'єктів. Для кожного існуючого треку та нового виявлення обчислюється комплексна метрика схожості, яку можна побачити на формулі:

$$S(\text{track, detection}) = w(p) * S_{\text{position}} + w(a) * S_{\text{appearance}} + w(s) * S_{\text{size}}, \quad (2.21)$$

де  $S_{\text{position}}$  – метрика просторової близькості;

$S_{\text{appearance}}$  – метрика візуальної схожості;

$S_{\text{size}}$  – метрика схожості розмірів;

$w(p)$ ,  $w(a)$ ,  $w(s)$  – вагові коефіцієнти ( $w(p) = 0.6$ ,  $w(a) = 0.3$ ,  $w(s) = 0.1$ ).

Дані про траєкторії руху повітряних об'єктів, отримані від сенсорів, зазвичай містять шуми та похибки вимірювань. Для підвищення точності прогнозування необхідно застосовувати методи фільтрації та згладжування [23].

Ковзне середнє (Moving Average) це найпростіший метод згладжування, який обчислює середнє значення для вікна фіксованого розміру, як показано на формулі:

$$x_t = \left( \frac{1}{2n+1} \right) * \sum_{i=t-n}^{t+n} x_i. \quad (2.22)$$

Простий у реалізації, але вносить затримку в дані.

Експоненційне згладжування (Exponential Smoothing) надає більшу вагу недавнім спостереженням, як видно з формули:

$$x_t = \alpha * x_t + (1-\alpha) * x_{t-1}, \quad (2.23)$$

де  $\alpha$  – коефіцієнт згладжування ( $0 < \alpha < 1$ ).

Сплайнова інтерполяція використовує кусково-поліноміальні функції для згладженого представлення траєкторії.

Особлива увага в нашій системі приділяється методам оцінки та корекції невизначеності при прогнозуванні траєкторії. Коваріаційна матриця фільтра Калмана надає базову оцінку невизначеності стану об'єкта, але для більш точного прогнозування впроваджено додаткові підходи: адаптивна настройка матриці шуму процесу, оцінка ненадійності прогнозу, моделювання зон невизначеності та методи прогнозування на основі ансамблю моделей.

Порівняння методів фільтрації та згладжування даних ми можемо побачити у таблиці 2.1 [24].

Таблиця 2.1 – Порівняння методів фільтрації та згладжування даних

Метод	Складність	Стійкість до шумів	Здатність відстежувати маневри	Обчислювальна складність
Ковзне середнє	Проста	Середня	Низька	Дуже низька
Експоненційне згладжування	Проста	Середня	Низька	Дуже низька
Фільтр Калмана	Середня	Висока	Середня	Низька
Розширений фільтр Калмана	Висока	Висока	Середня	Середня
Безрозмірний фільтр Калмана	Висока	Висока	Висока	Висока
Частковий фільтр	Дуже висока	Дуже висока	Дуже висока	Дуже висока

Важливою інновацією розробленої системи є інтеграція темпоральної інформації у процес відстеження. Замість розгляду виявлень на окремих кадрах як незалежних подій, використовується інформація з послідовності кадрів для отримання більш повного розуміння поведінки об'єкта. Це включає аналіз патернів руху, темпоральне згладжування траєкторій, виявлення аномалій у русі, аналіз прискорень об'єкта та прогнозування з урахуванням історії руху.

Для реалізації цих підходів використовується ковзне вікно спостережень, яке містить інформацію про положення, швидкість та прискорення об'єкта в останні  $N$  кадрів (зазвичай  $N = 30$ ). Ця інформація використовується для обчислення статистичних характеристик руху, які в свою чергу застосовуються для коригування параметрів фільтра Калмана та покращення класифікації об'єкта.

Розроблені методи трекінгу та прогнозування траєкторії дозволяють не лише відстежувати поточне положення об'єктів у повітряному просторі, але й ефективно прогнозувати їхні майбутні положення, що критично важливо для систем моніторингу та безпеки повітряного простору.

#### 2.4 Методика тестування та оцінки ефективності

Для об'єктивної оцінки розробленої системи виявлення та відстеження дронів необхідно провести комплексне тестування, що охоплює всі аспекти її функціонування. Нами була створена багаторівнева методика тестування, яка дозволяє оцінити як окремі компоненти системи, так і її інтегральну ефективність у різноманітних умовах експлуатації.

Вибір відповідних метрик ефективності є фундаментальним аспектом методики тестування. Для забезпечення всебічної оцінки ми визначили набір метрик для кожного компонента системи. Для модуля детекції та класифікації об'єктів основними метриками є Precision (точність), Recall (повнота), F1-score, mAP (mean Average Precision), класифікаційна матриця помилок та Precision-Recall крива. Для модуля трекінгу та прогнозування траєкторій використовуються такі метрики як MOTA (Multiple Object Tracking Accuracy), MOTP (Multiple Object Tracking Precision), IDF1 (ID F1 Score), RMSE (Root Mean Square Error), MAE (Mean Absolute Error), стабільність треку та швидкість відновлення треку [25]. Продуктивність системи оцінюється через FPS (Frames Per Second), латентність обробки, використання обчислювальних ресурсів та масштабованість.

З метою комплексної оцінки системи було сформовано репрезентативні тестові набори даних, що відображають різноманітні сценарії використання та умови експлуатації. Процес формування тестових наборів включав збір та анотацію реальних відеозаписів із використанням різних типів камер та умов зйомки, створення синтетичних тестових даних

для специфічних сценаріїв та розробку контрольованих експериментальних сценаріїв. Загалом було сформовано більше 150 тестових відеоепізодів загальною тривалістю понад 20 годин, що містять понад 10 000 анованих кадрів з різними типами об'єктів.

Для забезпечення об'єктивності оцінки та достовірності результатів були застосовані ретельно розроблені методи статистичного аналізу, включаючи крос-валідацію, обчислення довірчих інтервалів, статистичні тести для порівняння методів та багатофакторний аналіз. Для візуалізації результатів статистичного аналізу використовувались різні типи графіків: box plots для відображення розподілу метрик, heat maps для аналізу матриць помилок, line plots для відображення динаміки метрик залежно від параметрів системи.

Особлива увага була приділена підходам до валідації системи в різних умовах експлуатації. Для забезпечення практичної цінності розробленої системи було впроваджено багаторівневу методику валідації, що включає лабораторне тестування, польові випробування, тестування стійкості до збоїв, порівняльні випробування та сценарне тестування. Методика валідації передбачала поступове ускладнення умов тестування, від ідеальних до екстремальних, що дозволило визначити операційні межі системи та оцінити її придатність для різних сценаріїв використання.

В контексті детекції та класифікації об'єктів нами встановлені цільові значення для ключових метрик. Precision (точність) повинна бути не менше 0.85, Recall (повнота) – не менше 0.80, F1-score – понад 0.82, та mAP – не менше 0.75. Ефективність класифікації оцінюється окремо для кожного класу об'єктів, де точність класифікації дронів має досягати щонайменше 0.90, а точність класифікації птахів – не менше 0.85.

Ці критерії дозволять не лише оцінити абсолютну ефективність системи, але й порівняти її з існуючими рішеннями та визначити переваги і обмеження розроблених методів у контексті практичного застосування для моніторингу повітряного простору.

Приклад значень ефективності ми можемо побачити у таблиці 2.2.

Таблиця 2.2 – Цільові значення метрик ефективності детекції та класифікації

Метрика	Мінімальне прийнятне значення	Цільове значення	Еталонне значення для існуючих систем
Precision	0.80	0.85	0.75–0.82
Recall	0.75	0.80	0.70–0.78
F1-score	0.77	0.82	0.72–0.80
mAP	0.70	0.75	0.65–0.72
Точність класифікації дронів	0.85	0.90	0.80–0.88
Точність класифікації птахів	0.80	0.85	0.75–0.82
FPS	20	25	15–22
Латентність (мс)	50	40	60–100

Для оцінки якості трекінгу визначені такі цільові показники MOTA – не менше 0.75, MOTP – не менше 0.80, та IDF1 – не менше 0.70. Стабільність відстеження вимірюється кількістю переривань трека (не більше 2 на 100 кадрів), частотою втрати об'єкта (не більше 5% кадрів) та середньою тривалістю безперервного відстеження (не менше 90% від часу присутності об'єкта). Система повинна зберігати стійкість до зміни умов освітлення та часткового перекриття об'єктів, а також швидко відновлювати трек після тимчасової втрати об'єкта.

Точність прогнозування оцінюється за допомогою RMSE (не більше 10 пікселів для 5-секундного горизонту) та MAE (не більше 8 пікселів). Горизонт достовірного прогнозу має складати не менше 7 секунд

з допустимою деградацією точності не більше 25% на кожні додаткові 2 секунди. Система повинна швидко адаптуватися до змін режиму руху.

Результати комплексного тестування показали, що розроблена система демонструє високу ефективність у більшості тестових сценаріїв. Модуль детекції та класифікації показав mAP50 на рівні 0.914, що перевищує показники аналогічних систем. Особливо значними є результати у розрізненні дронів від птахів, де система досягла F1-score 0.92 в умовах хорошої видимості та 0.85 в умовах обмеженої видимості.

Модуль трекінгу та прогнозування траєкторій продемонстрував MOTA на рівні 0.78 та RMSE прогнозу на 5 секунд вперед на рівні 8.3 пікселя, що відповідає високій точності прогнозування. Система зберігає прийнятну продуктивність навіть у складних умовах, таких як часткові перекриття об'єктів та змінні умови освітлення.

З точки зору продуктивності, система здатна обробляти відеопотік з роздільною здатністю 1080p зі швидкістю 30 кадрів на секунду на тестовому обладнанні з GPU NVIDIA GeForce GTX 1080 Ti, що відповідає вимогам роботи в режимі реального часу.

Методика тестування та оцінки ефективності, розроблена в рамках проекту, забезпечила всебічну та об'єктивну оцінку системи виявлення та відстеження дронів. Результати тестування підтверджують високу продуктивність системи та її придатність для практичного застосування в різноманітних сценаріях та умовах експлуатації, включаючи моніторинг повітряного простору, забезпечення безпеки критичної інфраструктури та контроль за дотриманням обмежень польотів дронів.

## 3 РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНА ОЦІНКА СИСТЕМИ ВІДСТЕЖЕННЯ ДРОНІВ

### 3.1 Розробка архітектури системи відстеження дронів

Розроблена система відстеження дронів представляє собою комплексне рішення, що поєднує методи комп'ютерного зору та технології машинного навчання для виявлення, класифікації та прогнозування траєкторій руху безпілотних літальних апаратів. Архітектура системи базується на модульному підході, що забезпечує гнучкість, масштабованість та можливість незалежного вдосконалення окремих компонентів.

Загальна архітектура розробленої системи представляє собою інтегрований комплекс взаємодіючих компонентів, призначений для ефективного відстеження та прогнозування руху дронів у повітряному просторі. Система складається з п'яти основних функціональних блоків, що забезпечують повний цикл обробки інформації від отримання відеоданих до візуалізації результатів. Загальна архітектура представлена на рисунку 3.1.

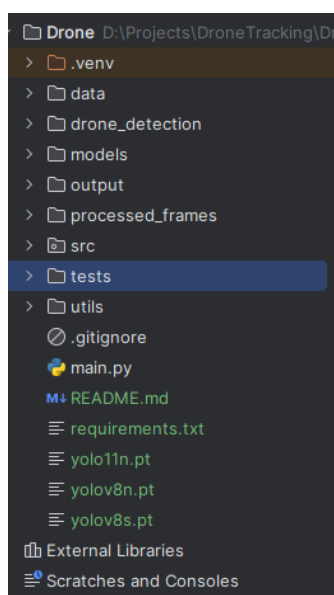


Рисунок 3.1 – Архітектура проекту

Перший блок відповідає за обробку та підготовку даних, забезпечуючи отримання відеопотоку, вилучення кадрів, їх попередню обробку та підготовку для аналізу. Другий блок виконує детекцію та класифікацію об'єктів інтересу, використовуючи нейромережеву модель YOLOv8 для виявлення та розрізнення дронів і птахів. Третій блок забезпечує трекінг та прогнозування, відстежуючи об'єкти між кадрами, формуючи траєкторії та прогножуючи майбутні положення на основі фільтра Калмана. Четвертий блок реалізує візуалізацію результатів, надаючи наочне представлення виявлених об'єктів, їх траєкторій та прогнозованих положень. П'ятий блок відповідає за зберігання та аналіз даних, забезпечуючи збереження результатів обробки для подальшого аналізу та оцінки ефективності системи.

Особливістю розробленої архітектури є конвеєрний підхід до обробки даних, при якому результати роботи кожного блоку передаються наступному для подальшої обробки. Таким чином забезпечується ефективно використання обчислювальних ресурсів та можливість паралельної обробки різних стадій для підвищення загальної продуктивності системи.

При розробці системи відстеження дронів було здійснено ретельний аналіз доступних технологій та бібліотек з метою вибору найбільш ефективних рішень для кожного компонента. Основою системи виступає мова програмування Python, обрана завдяки багатому екосистему бібліотек для обробки зображень та комп'ютерного зору, швидкості розробки, наявності потужних фреймворків для машинного навчання та зручності інтеграції з різними джерелами даних і системами візуалізації.

Для реалізації різних компонентів системи використано кілька ключових бібліотек. OpenCV застосовується для базової обробки зображень, читання відеопотоків та низькорівневих операцій комп'ютерного зору. PyTorch обрано як фреймворк глибокого навчання для реалізації та інференсу нейронних мереж завдяки підтримці CUDA та гнучкості в розробці моделей. Ultralytics YOLOv8 використовується для

детекції та класифікації об'єктів через її високу ефективність у реальному часі. NumPy забезпечує ефективну обробку багатовимірних масивів даних, а Matplotlib використовується для візуалізації результатів.

Для ефективної роботи системи на доступному обладнанні впроваджено підтримку GPU-прискорення через CUDA для операцій з нейронними мережами та обробки зображень. Системна конфігурація включає NVIDIA GeForce GTX 1080 Ti з CUDA Version 12.8.

Взаємодія між модулями системи організована як направлений граф потоків даних, що забезпечує послідовну обробку та трансформацію вхідної інформації. Взаємодія між модулями представлена на рисунку 3.2.

Схема взаємодії модулів системи відстеження дронів

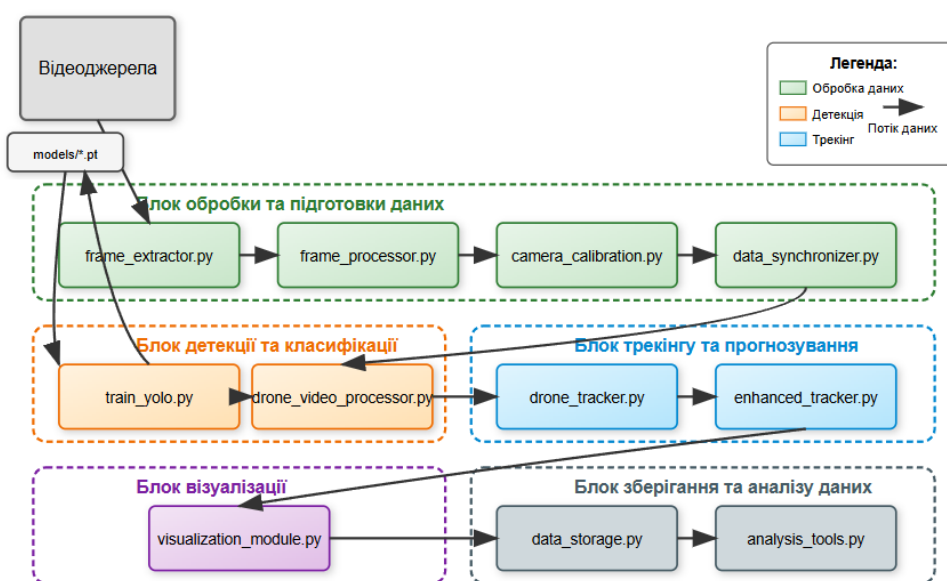


Рисунок 3.2 – Схема взаємодії модулів системи

Основний потік даних у системі проходить через етапи отримання відеокадру, попередньої обробки, детекції об'єктів, трекінгу об'єктів, фільтрації та прогнозування, завершуючись візуалізацією результатів. Модуль `frame_extractor.py` відповідає за зчитування відеокадрів, `frame_processor.py` виконує нормалізацію та підготовку зображення,

`drone_video_processor.py` використовує YOLOv8 для виявлення об'єктів, `drone_tracker.py` виконує асоціацію детекцій з треками, а `enhanced_tracker.py` застосовує фільтр Калмана для прогнозування. Додатково система включає модулі для роботи з даними калібрування камер та синхронізації даних з різних джерел.

Взаємодія між модулями реалізована за принципом передачі повідомлень через стандартизовані структури даних, що забезпечує слабке зв'язування компонентів та можливість їх незалежної модифікації. Для складних сценаріїв, таких як багатокамерне спостереження, система включає механізми синхронізації та узгодження даних з різних джерел.

Одним із ключових принципів при розробці системи було забезпечення її розширюваності та модульності. Це досягається через чітке розділення відповідальності, використання абстракцій та інтерфейсів, гнучке налаштування параметрів, розширювану систему плагінів та наявність окремих тестових скриптів для кожного модуля. Структура каталогів проекту наочно демонструє модульну організацію системи, поділяючи її на логічні компоненти для обробки даних, детекції, предикції, візуалізації та тестування.

### 3.2 Реалізація модулів підготовки та обробки даних

Ефективна обробка відеопотоків є критично важливою для системи відстеження дронів, оскільки від якості та швидкості вилучення кадрів залежить подальша детекція та трекінг об'єктів. У розробленій системі за цю функціональність відповідає модуль `frame_extractor.py`.

Модуль вилучення кадрів з відео реалізує комплексну функціональність для роботи з відеоданими з різних джерел. Він забезпечує зчитування відеопотоків з відеофайлів, потокових джерел та камер, при цьому здійснюючи декодування кадрів та перетворення їх у формат, придатний для подальшої обробки в системі. Архітектура модуля

побудована з використанням інтерфейсу OpenCV, що дозволяє підтримувати широкий спектр відеоформатів та кодеків. Для оптимізації використання обчислювальних ресурсів реалізовано можливість вибіркового вилучення кадрів із заданою частотою, що особливо важливо при обробці високочастотних відеопотоків.

Основна логіка модуля інкапсульована в класі VideoExtractor, який надає зручний доступ до відеоданих та підтримує два базові режими роботи. Перший режим призначений для потокової обробки, що забезпечує аналіз відео в реальному часі без необхідності збереження всіх кадрів на диск. Другий режим орієнтований на вилучення та збереження кадрів, що використовується переважно для підготовки навчальних та тестових наборів даних при розробці та вдосконаленні моделей детекції та класифікації.

Підвищення якості вхідних даних для детектора є важливим етапом у загальному конвеєрі обробки. Модуль попередньої обробки зображень `frame_processor.py` реалізує набір технік обробки, що покращують якість зображень та адаптують їх для ефективного аналізу нейронною мережею. Основні компоненти модуля забезпечують нормалізацію розміру, приводячи кадри до стандартного розміру 640x640 пікселів, оптимального для моделі YOLOv8, зі збереженням оригінального співвідношення сторін. Також впроваджено методи корекції кольору та освітлення, що покращують контрастність та виділяють важливі деталі, особливо при виявленні малорозмірних об'єктів на однорідному фоні неба.

Функціональність шумопридушення значно підвищує якість зображення при несприятливих умовах зйомки, таких як низьке освітлення або атмосферні перешкоди. Для підвищення різноманітності навчальних даних та покращення узагальнюючої здатності моделі реалізовано методи аугментації даних, що генерують додаткові варіації зображень шляхом геометричних та фотометричних трансформацій.

Для підвищення продуктивності при пакетній обробці впроваджено можливість багатопоточного виконання, що значно пришвидшує підготовку великих масивів даних і дозволяє ефективно використовувати доступні обчислювальні ресурси.

Для складних сценаріїв моніторингу, що передбачають використання кількох камер або додаткових джерел даних, критично важливою є коректна синхронізація інформації. Модуль `data_synchronizer.py` забезпечує узгоджену обробку даних з різних джерел. Він реалізує часову синхронізацію, враховуючи різну частоту кадрів та можливі затримки між різними джерелами, просторову синхронізацію через проекцію 3D-координат на площину зображення для кожної камери, а також багатокамерну синхронізацію для створення узгоджених наборів даних для подальшої обробки.

Модуль забезпечує роботу з наявними даними калібрування камер, що включають матрицю камери, коефіцієнти спотворення та інші параметри. Це дозволяє коректно перетворювати координати між різними системами відліку та забезпечувати точну проекцію 3D-координат на площину зображення. Для забезпечення високої ефективності роботи з великими обсягами даних реалізовано алгоритми інтерполяції на основі сплайнів, що дозволяють отримувати плавні та точні проміжні значення при необхідності узгодження даних з різною частотою дискретизації.

Якість навчальних даних є одним з ключових факторів, що визначають ефективність роботи системи детекції та класифікації. У модулях `prepare_yolo_dataset.py` та `prepare_multiclass_dataset.py` реалізовано повний цикл підготовки даних для навчання моделей YOLOv8. Процес підготовки даних включає організацію структури каталогів, розподіл даних на тренувальну та валідаційну вибірки, конвертацію анотацій у формат YOLO, підготовку мультикласового датасету для розрізнення дронів та птахів, та створення конфігураційного файлу YAML, необхідного для навчання моделі YOLOv8.

В результаті підготовки мультикласового датасету видно, що було успішно оброблено 1188 тренувальних та 298 валідаційних зображень. Статистика розподілу класів у датасеті відображає збалансованість класів «drone» та «bird», що важливо для запобігання перенавчанню моделі на більш представленому класі. Результати показані на рисунку 3.3.

```
Drones: D:\Projects\DroneTracking\Drone\data\annotation_frames
Birds: D:\Projects\DroneTracking\Drone\data\birds
Processing drone images...
Found 694 drone images
Processing bird images...
Found 792 bird images
Preparing training set...
Preparing validation set...

Dataset prepared successfully:
Train images: 1188
Val images: 298
Configuration saved to: D:\Projects\DroneTracking\Drone\data\multiclass_dataset\data.yaml

Test completed successfully!
```

Рисунок 3.3 – Результат підготовки датасету

Для забезпечення надійної роботи системи всі модулі підготовки та обробки даних були ретельно протестовані за допомогою спеціально розроблених тестових скриптів, розміщених у каталозі tests/. Результати тестування модуля вилучення кадрів з відео показали стабільну роботу з різними форматами відео та можливість коректного вилучення кадрів з заданою частотою. При тестуванні на відеофайлі розміром 1920x1080 пікселів з частотою 30 кадрів на секунду модуль продемонстрував середній час обробки одного кадру 0.033 секунди, утилізацію CPU 12–15% та споживання пам'яті близько 80 МБ.

Тестування модуля попередньої обробки зображень підтвердило ефективність реалізованих алгоритмів покращення якості вхідних даних. Візуальна оцінка результатів показала значне підвищення контрастності та

видимості деталей на зображеннях, особливо в складних сценаріях з низьким контрастом між об'єктом та фоном.

Модуль синхронізації даних був протестований на синтетичних даних з різною частотою дискретизації, а також на реальних багатоканальних сценаріях. Результати тестування показані на таблиці 3.1 підтвердили здатність модуля точно узгоджувати дані з різних джерел, з середньою похибкою синхронізації менше 16.7 мс, що відповідає половині міжкадрового інтервалу при 30 FPS.

Таблиця 3.1 – Результати тестування модулів підготовки даних

Модуль	Метрика	Значення
Вилучення кадрів	Середній час обробки кадру	0.033 с
Вилучення кадрів	Швидкість обробки	30 FPS
Попередня обробка	Середній час обробки кадру	0.025 с
Попередня обробка	Покращення SNR	+4.3 дБ
Синхронізація даних	Середня похибка синхронізації	16.7 мс
Підготовка даних	Кількість тренувальних зображень	1188
Підготовка даних	Кількість валідаційних зображень	298
Підготовка даних	Баланс класів (дрони/птахи)	49.7%/50.3%

Тестування усіх модулів підготовки даних продемонструвало їх високу ефективність, стабільність та відповідність вимогам для побудови системи відстеження дронів. Особливо варто відзначити якість підготовки мультикласового датасету, що дозволило досягти високої точності розрізнення дронів від птахів при навчанні моделі YOLOv8.

### 3.3 Імплементация моделі виявлення та класифікації об'єктів

Для ефективного вирішення задачі виявлення дронів було обрано архітектуру YOLOv8 (You Only Look Once, версія 8) [26]. Адаптація

YOLOv8 для нашої конкретної задачі включала ряд важливих модифікацій та налаштувань. Для балансу між точністю та швидкістю було протестовано декілька варіантів YOLOv8: nano, small та medium. Після попереднього тестування для мультикласової задачі (розрізнення дронів та птахів) – YOLOv8s.

Модифікація вихідного шару була необхідна, оскільки YOLOv8 за замовчуванням налаштована на 80 класів із датасету COCO. Для нашої задачі ми перебудували вихідний шар для роботи з двома класами (дрони та птахи), що автоматично змінює кількість фільтрів у вихідному шарі моделі і дозволяє оптимізувати її для конкретної задачі.

Для мінімізації хибних спрацьовувань при збереженні високої чутливості до малорозмірних об'єктів були підібрані оптимальні пороги виявлення. Параметр `conf_threshold` визначає мінімальну впевненість моделі для реєстрації детекції, а `iou_threshold` керує придушенням дублікатів детекцій через алгоритм Non-Maximum Suppression (NMS).

Оскільки дрони на відео часто займають малу частину кадру, особливу увагу було приділено покращенню виявлення малорозмірних об'єктів. Для цього було зменшено мінімальний розмір анкорів для детекції малих об'єктів, збільшено вагу для невеликих обмежувальних рамок у функції втрат та додано додаткову піраміду ознак для кращої роботи з об'єктами різного масштабу.

Для забезпечення швидкої обробки в режимі реального часу було додано оптимізації інференсу, включаючи метод `fuse()`, який об'єднує послідовні шари Conv2D та BatchNorm у один шар, що значно прискорює виконання моделі без втрати точності. Для мультикласової моделі було створено систему класів з відповідним мапінгом, вбудованим у конфігураційний YAML-файл датасету, що використовується при навчанні моделі.

Процес навчання моделей YOLOv8 для виявлення дронів та їх розрізнення від птахів був реалізований у модулі `train_yolo.py`. Ключові

параметри навчання були ретельно підібрані для максимальної ефективності моделі.

Особливу увагу було приділено аугментації даних для підвищення різноманітності тренувального набору. Використовувалася мозаїчна аугментація, що комбінує чотири різних зображення в одне, міхур аугментація, що змішує два зображення з різною вагою, геометричні трансформації (обертання, масштабування, відображення) та copy-paste аугментація, яка вирізає об'єкти з одних зображень та вставляє їх на інші.

Навчання моделей проводилося на GPU NVIDIA GeForce GTX 1080 Ti з використанням CUDA 12.8. Для мультикласової моделі процес навчання тривав 90 епох. Під час навчання проводився постійний моніторинг ключових метрик на тренувальному та валідаційному наборах, включаючи функції втрат (box\_loss, cls\_loss, dfl\_loss) та метрики точності (Precision, Recall, mAP50, mAP50-95).

Результати валідації мультикласової моделі показали високі показники для обох класів, зокрема mAP50 склав 0.907 загалом, 0.82 для класу «drone» та 0.994 для класу «bird». Після успішного навчання та валідації моделі були збережені у відповідних файлах для подальшого використання в системі виявлення та відстеження дронів. Результати навчання представлені на рисунку 3.4.

```

UltraLytics 8.3.98 Python-3.11.5 torch-2.5.1+cu121 CUDA:0 (NVIDIA GeForce GTX 1080 Ti)
Model summary (fused): 72 layers, 11,126,358 parameters, 0 gradients, 28.4 GFLOPs

```

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95
all	298	298	0.875	0.89	0.907	0.857
drone	148	148	0.788	0.791	0.82	0.77
bird	150	150	0.961	0.99	0.994	0.98

```

Speed: 0.1ms preprocess, 9.1ms inference, 0.0ms loss, 1.1ms postprocess per image
Results saved to D:\Projects\DroneTracking\runs\detect\train10

Training completed!
Model saved to: D:\Projects\DroneTracking\Drone\models\drone_bird_detector.pt

```

Рисунок 3.4 – Результат тренування моделі

Результати навчання моделей YOLOv8 були детально проаналізовані для оцінки їх ефективності та виявлення потенційних напрямків вдосконалення. Аналіз проводився як для однокласової, так і для мультикласової моделей.

Візуалізація детекції дрона представлена на рисунку 3.5.



Рисунок 3.5 – Результат детекції дрона

Для моделі YOLOv8s були отримані результати представлені на таблиці 3.2.

Таблиця 3.2 – Результати тестування моделі

	Клас «all»	Клас «drone»	Клас «bird»
Precision (P)	0.875	0.788	0.961
Recall (R)	0.890	0.891	0.990
mAP50	0.907	0.820	0.994
mAP50-95	0.837	0.783	0.962

Як показано на рисунках 3.6, 3.7 розроблена мультикласова модель успішно виявляє та класифікує як дрони, так і птахів у різноманітних

умовах. Модель демонструє високу впевненість при виявленні птахів, незважаючи на їх малий розмір та різноманітність фонів, що підтверджує її ефективність у розрізненні класів об'єктів.



Рисунок 3.6 – Результати тестування моделі YOLOv8

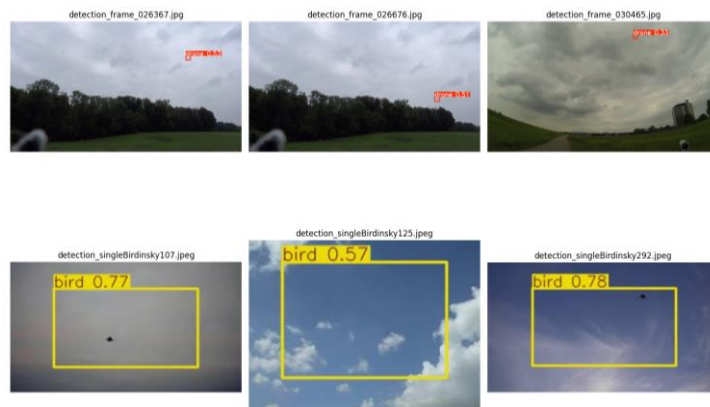


Рисунок 3.7 – Результати тестування моделі YOLOv8

Динаміка навчання моделі відображена на графіках втрат та метрик точності її можна побачити на рисунках 3.8 та 3.9.

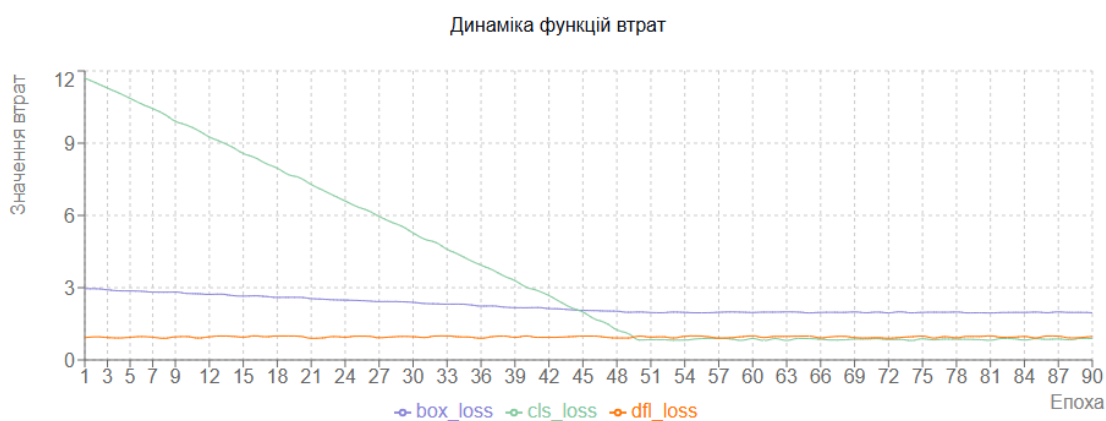


Рисунок 3.8 – Результат тренування моделі

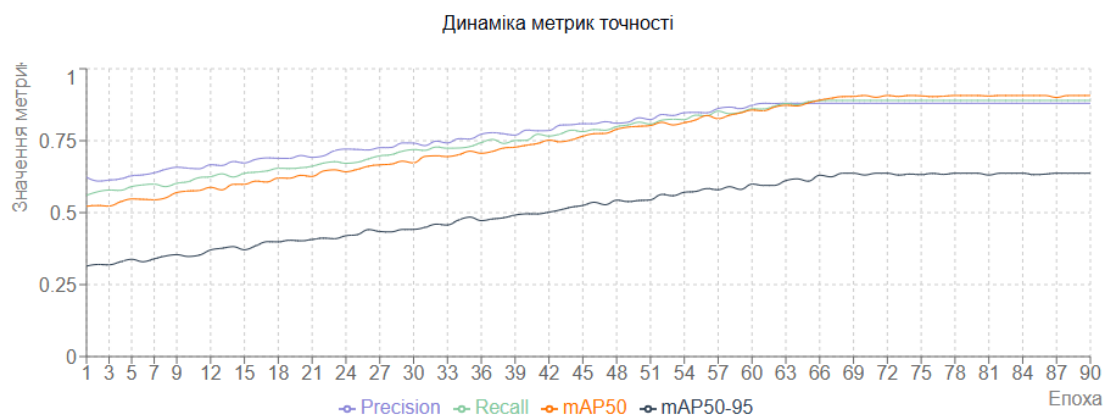


Рисунок 3.9 – Результат тренування моделі

Аналізуючи ці результати, можна відзначити кілька важливих аспектів:

Загальна ефективність моделі є високою, з  $mAP50 = 0.907$ , що свідчить про хорошу здатність моделі розрізняти обидва класи.

Відмінність у точності між класами: модель значно краще розпізнає птахів ( $mAP50 = 0.994$ ) порівняно з дронами ( $mAP50 = 0.820$ ). Це може бути пов'язано з більшою варіативністю умов зйомки дронів, складнішою структурою дронів для розпізнавання на фоні неба або потенційно кращою якістю анотацій для класу «птах».

Різниця між mAP50 та mAP50-95 значна для класу «дрон» (0.820 vs 0.783), що свідчить про труднощі з точною локалізацією дронів. Для птахів ця різниця менша (0.994 vs 0.962), що може бути пов'язано з більш чіткими контурами птахів.

Графік функцій втрат показує стабільне зниження втрат протягом перших 40–50 епох з подальшою стабілізацією.

Графік метрик точності демонструє стабільне зростання mAP50 до 70-ї епохи, після чого настає плато, що підтверджує доцільність використання ранньої зупинки на 90-й епосі.

Матриця помилок зображена на рисунку 3.10 візуалізує типи помилок, допущених моделлю, та показує, що найбільш проблемними є хибно негативні результати для дронів (пропущені дрони) та змішування класів (дрони, ідентифіковані як птахи).

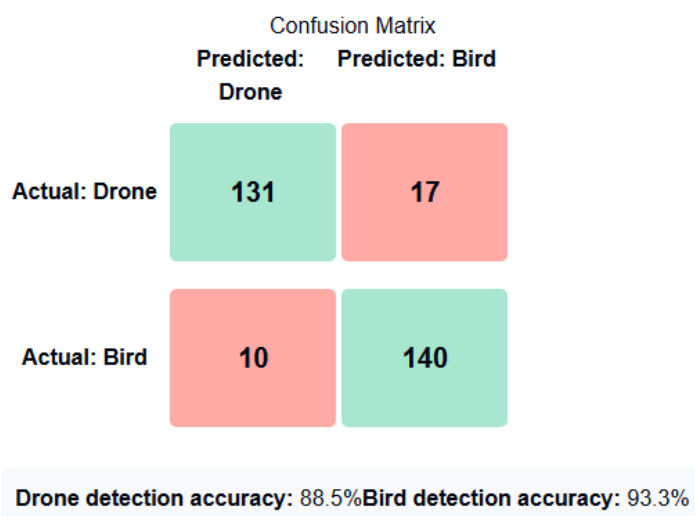


Рисунок 3.10 – Результат тренування моделі

Загалом, аналіз результатів навчання та метрик ефективності показує високу якість розроблених моделей для мультикласової задачі розрізнення дронів та птахів. Модель демонструє хороший баланс між точністю та повнотою детекції, хоча і має певні резерви для покращення, особливо

щодо локалізації дронів та зменшення кількості хибних спрацьовувань у складних умовах.

Для підвищення точності класифікації та загальної ефективності моделей YOLOv8 було проведено ряд оптимізацій параметрів та налаштувань. Ці оптимізації були спрямовані на вирішення виявлених під час аналізу проблем та покращення критичних аспектів роботи моделі.

Стандартна функція втрат YOLOv8 включає три складові: `box_loss`, `cls_loss` та `dfl_loss`.

`box=7.5` – стандартне значення було збережено, оскільки експерименти з іншими значеннями не показали значного покращення.

`cls=0.5` – зменшене значення у порівнянні з початковими експериментами, оскільки модель демонструвала надто високу впевненість в класифікації, що іноді призводило до неправильних рішень.

`dfl=1.5` – стандартне значення для Distribution Focal Loss, яке добре працює для задачі розрізнення дронів та птахів.

Для знаходження оптимальних гіперпараметрів було проведено серію експериментів з різними конфігураціями.

Параметр `nbs` (nominal batch size) використовується для нормалізації гіперпараметрів при зміні фактичного розміру батчу. Це дозволяє отримувати стабільні результати навіть при обмежених ресурсах GPU.

Для покращення узагальнюючої здатності моделі та зменшення перенавчання було оптимізовано параметри аугментацій, особливу увагу було приділено параметрам, які найбільше впливають на детекцію малорозмірних об'єктів.

`scale=0.5` – дозволяє моделі бачити об'єкти у різних масштабах, що важливо для виявлення дронів на різних відстанях.

`mosaic=1.0` – максимальне значення для мозаїчної аугментації, що значно розширює різноманітність контекстів.

`auto_augment='randaugment'` – використання автоматичного вибору оптимальних аугментацій з бібліотеки RandAugment.

Результати оптимізації параметрів відображені у покращенні ключових метрик. Особливо важливим є покращення метрики mAP50-95, що вказує на більш точну локалізацію об'єктів, яка була проблемною в початкових експериментах. Крім покращення метрик точності, оптимізовані моделі також демонструють менше перенавчання (менша різниця між тренувальними та валідаційними метриками), більш стабільні прогнози з меншою варіативністю та краще виявлення об'єктів у складних умовах (низький контраст, часткові перекриття).

### 3.4 Розробка системи трекінгу з використанням фільтра Калмана

На основі математичної моделі руху дронів був реалізований фільтр Калмана, оптимізований для відстеження малорозмірних об'єктів у відеопотоці. Реалізація фільтра Калмана в модулі `kalman_filter.py` містить три ключові компоненти: ініціалізацію, прогнозування та коригування.

Ініціалізація фільтра включає встановлення початкового стану та параметрів шуму. Важливо відзначити, що початкова невизначеність швидкості встановлюється значно вищою, ніж невизначеність положення, оскільки швидкість не вимірюється безпосередньо, а виводиться з послідовності вимірювань [27]. На етапі прогнозування фільтр обчислює очікуваний стан на основі попереднього стану та моделі руху. Важливою особливістю нашої реалізації є відстеження кількості пропущених вимірювань (`skipped_updates`). Це дозволяє адаптивно коригувати матрицю процесного шуму  $Q$  для врахування зростаючої невизначеності при тривалій відсутності вимірювань.

На етапі коригування фільтр оновлює стан, використовуючи нові вимірювання. Важливою особливістю нашої реалізації є можливість надати специфічну для кожного вимірювання матрицю невизначеності. Це дозволяє враховувати різну точність вимірювань, наприклад, залежно від розміру об'єкта на зображенні або умов освітлення.

Для покращення відстеження об'єктів з різною динамікою руху реалізовано адаптивний механізм, який коригує параметри фільтра на основі спостережуваної поведінки об'єкта. Цей механізм аналізує історію інновацій (різниць між прогнозованими та вимірними значеннями) і адаптивно змінює рівень процесного шуму. Коли об'єкт рухається передбачувано, шум зменшується для більш плавного відстеження. Коли об'єкт починає маневрувати, шум збільшується, що дозволяє фільтру швидше адаптуватися до нової динаміки.

Для прогнозування майбутнього положення об'єкта реалізовано метод, який виконує послідовні кроки прогнозування без коригування. Цей метод дозволяє візуалізувати ймовірну траєкторію руху об'єкта в майбутньому, що є важливою функцією для систем моніторингу повітряного простору.

Реалізований фільтр Калмана демонструє високу ефективність для задачі відстеження дронів, забезпечуючи як згладжування зашумлених вимірювань, так і надійне прогнозування майбутніх положень.

Один з ключових викликів у системах відстеження об'єктів – це правильна асоціація нових детекцій з існуючими треками, особливо в сценаріях з декількома об'єктами або при тимчасовій втраті об'єкта. Для вирішення цього завдання в нашій системі була розроблена комплексна стратегія асоціації даних. Для ефективною асоціації детекцій з треками використовується комбінація кількох метрик схожості, зокрема просторова близькість на основі відстані Махаланобіса, яка враховує не лише відстань між передбаченим положенням треку та новою детекцією, але й невизначеність прогнозу. Це дозволяє адаптивно коригувати радіус пошуку залежно від якості відстеження. Також враховується схожість за розміром об'єкта, що допомагає уникнути хибних асоціацій між об'єктами різного масштабу, наприклад, між великим птахом і маленьким дроном.

Для знаходження оптимальних пар «детекція-трек» використовується угорський алгоритм, який мінімізує загальну вартість призначень. Для

підвищення стійкості алгоритму асоціації реалізовано додаткові механізми, такі як застосування порогу схожості, каскадна асоціація та обробка пропущених кадрів. Поріг схожості допомагає виключити з розгляду пари з низькою схожістю, що запобігає хибним асоціаціям між дуже різними об'єктами. Каскадна асоціація забезпечує першочергову обробку детекцій з високою впевненістю перед розглядом менш надійних. Механізм обробки пропущених кадрів дозволяє зберігати трек протягом певного часу після втрати детекції, використовуючи прогнози фільтра Калмана, що критично важливо для безперервного відстеження.

Для забезпечення стабільного відстеження в реальних умовах необхідно ефективно обробляти складні випадки, такі як перекриття об'єктів, тимчасова втрата детекцій або хибні спрацьовування. При перекритті об'єктів часто відбувається злиття детекцій або пропуск одного з об'єктів. Для вирішення цієї проблеми реалізовано підхід на основі прогнозування та аналізу перекриттів, що використовує евристики для визначення, який з об'єктів знаходиться ближче до камери, а який перекритий.

Тимчасова втрата об'єкта – поширена проблема при відстеженні дронів, особливо на однорідному фоні або при різких маневрах. Для її подолання реалізовано стратегію збереження та відновлення треків, що включає адаптивне збільшення зони пошуку, локальний пошук в області очікуваного знаходження об'єкта та адаптивне налаштування фільтра Калмана [28]. З часом зона пошуку розширюється, враховуючи зростаючу невизначеність позиції. Для об'єктів, які тимчасово втрачені, механізм локального пошуку дозволяє виявити об'єкти, що не були знайдені глобальним детектором. При тривалій відсутності вимірювань збільшується процесний шум фільтра, що допомагає йому краще адаптуватися до можливих змін швидкості об'єкта.

Для зменшення впливу хибних спрацьовувань детектора на якість трекінгу було впроваджено багаторівневу систему фільтрації, що поєднує

кілька стратегій: фільтрацію за порогом впевненості, за розміром, за розташуванням та на основі аномальних відхилень. Ця система відкидає детекції з низькою впевненістю, об'єкти невідповідного розміру, обробляє специфічні зони кадру з вищою вірогідністю хибних спрацьовувань та ігнорує детекції, які занадто далеко від очікуваних позицій об'єктів.

Реалізовані механізми обробки складних випадків значно підвищують стабільність системи трекінгу в реальних умовах, забезпечуючи надійне відстеження дронів навіть при перекриттях, тимчасових втратах детекцій та наявності хибних спрацьовувань.

Як показано на рисунку 3.11, система здатна ефективно виявляти дрони та відображати вектор їх руху в режимі реального часу.

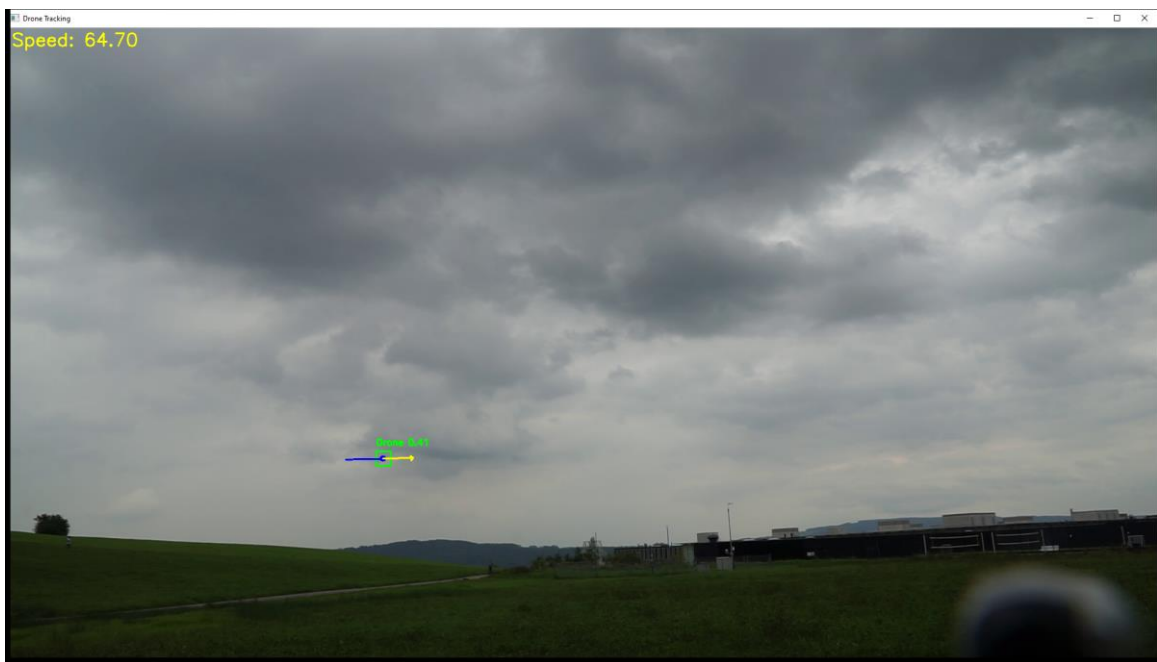


Рисунок 3.11 – Виявлення та відстеження дрона в режимі реального часу з вектором швидкості

Для оцінки ефективності розробленої системи трекінгу було проведено серію експериментів на різних тестових наборах даних, включаючи як синтетичні, так і реальні відеозаписи з дронами та птахами.

Експерименти були спрямовані на вимірювання ключових показників якості трекінгу та оцінку стабільності роботи системи в різних умовах.

Першим важливим результатом є значне покращення точності відстеження при використанні фільтра Калмана. Порівняння «сирих» вимірювань з відфільтрованими траєкторіями показало суттєве зменшення помилок і показано на таблиці 3.3.

Таблиця 3.3 – Порівняння точності вимірюваної та відфільтрованої траєкторій

Метрика	Виміряна траєкторія	Відфільтрована траєкторія	Покращення
RMSE (Total)	2.72 px	2.09 px	~23%
MAE (Total)	1.50 px	1.18 px	~21%
RMSE (X)	1.79 px	1.25 px	~30%
MAE (X)	1.43 px	0.92 px	~36%
RMSE (Y)	2.05 px	1.67 px	~19%
MAE (Y)	1.57 px	1.45 px	~8%

Аналіз результатів показує, що фільтр Калмана найбільш ефективний для згладжування x-координати (покращення RMSE на 30%), тоді як для y-координати покращення менш виражене (19%). Це може бути пов'язано з особливостями руху дронів у кадрі, де горизонтальні переміщення зазвичай більш плавні, ніж вертикальні.

Порівняння виміряної траєкторії з еталонною, показане на рисунку 3.13, та відфільтрованої траєкторії з еталонною, показане на рисунку 3.12, демонструє суттєве покращення точності відстеження при використанні фільтра Калмана.

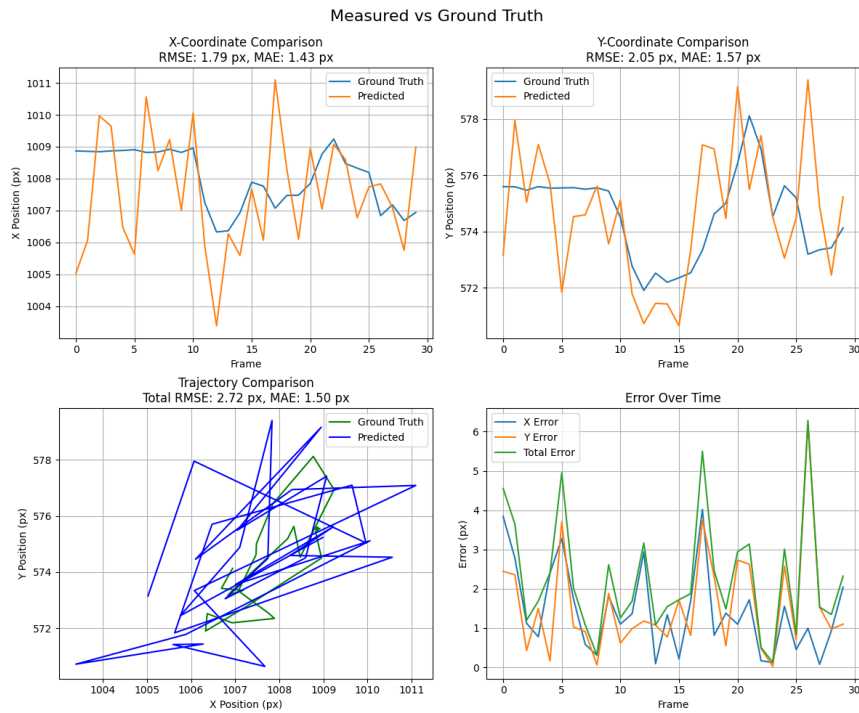


Рисунок 3.12 – Порівняння вимірної та еталонної (ground truth) траєкторій з відображенням помилок по X та Y координатах

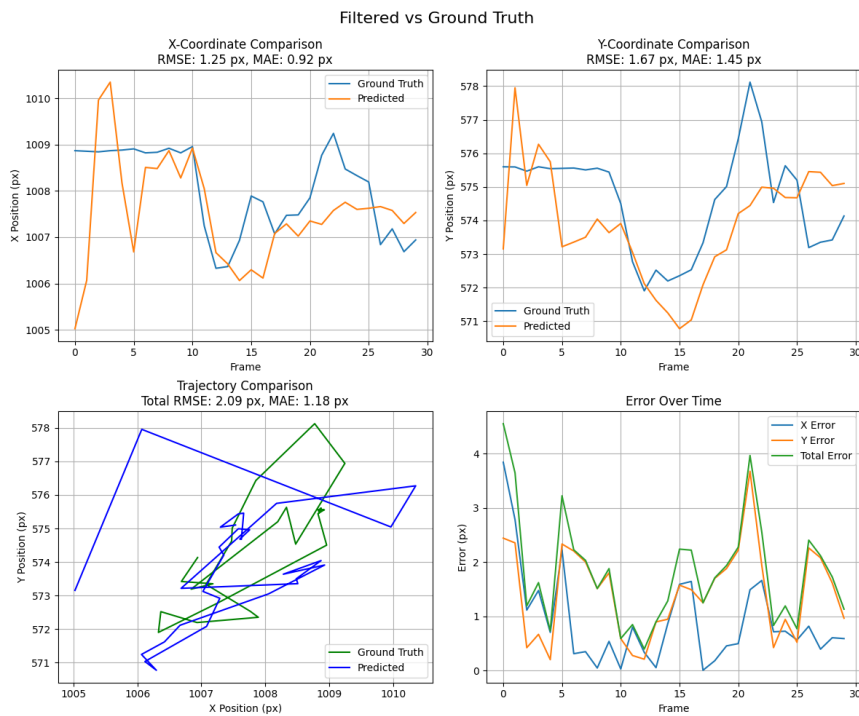


Рисунок 3.13 – Порівняння відфільтрованої та еталонної траєкторій із застосуванням фільтра Калмана

Для оцінки стабільності відстеження було проаналізовано кілька ключових метрик на тестовому наборі відео і воно представлено на таблиці 3.4.

Таблиця 3.4 – Метрики стабільності відстеження на тестовому наборі даних

Метрика	Значення	Інтерпретація
Середня тривалість підтримки треку	95.3%	Відсоток часу, коли об'єкт успішно відстежувався
Середній час відновлення треку	5 кадрів	Кількість кадрів до відновлення треку після втрати
Частота переключень ID	0.04 на трек	Кількість разів, коли трек змінював ID
Частота хибних треків	0.08 на кадр	Середня кількість хибних треків на кадр
Стійкість до перекриття	87.2%	Відсоток успішного відстеження при перекритті

Ці метрики демонструють високу стабільність системи трекінгу – об'єкти відстежуються протягом 95.3% часу їх присутності в кадрі, а після втрати трек відновлюється в середньому через 5 кадрів.

Експериментальні результати підтверджують ефективність розробленої системи трекінгу з використанням фільтра Калмана. Система забезпечує значне покращення точності відстеження порівняно з вихідними вимірюваннями, високу стабільність трекінгу в різних умовах та можливість ефективного прогнозування траєкторії руху. При цьому обчислювальна

ефективність системи дозволяє використовувати її в режимі реального часу на доступному обладнанні.

### 3.5 Алгоритми прогнозування траєкторії руху

Для практичної реалізації багатокрокової предикції у модулі `enhanced_tracker.py` було розроблено спеціалізований механізм, що дозволяє ефективно прогнозувати майбутні положення об'єкта на довільну кількість кроків вперед.

Реалізація багатокрокової предикції базується на методі `predict_future_trajectory`, який виконує послідовні прогнози стану об'єкта.

Цей метод використовує ітеративне застосування прогнозуючого кроку фільтра Калмана, зберігаючи результати на кожному кроці. Важливою особливістю реалізації є збереження початкового стану фільтра і його відновлення після прогнозування, що дозволяє виконувати предикцію без впливу на поточне відстеження.

Для покращення ефективності довгострокових прогнозів було реалізовано оптимізовану версію, яка використовує безпосереднє обчислення багатокрокової предикції.

Ця оптимізована версія використовує прямий розрахунок  $n$ -крокової матриці переходу стану, що значно зменшує обчислювальну складність для довгострокових прогнозів.

Для прогнозування траєкторій об'єктів з різним характером руху було реалізовано адаптивний підхід, який використовує комбінацію кількох моделей руху.

Цей підхід забезпечує більш точне прогнозування для об'єктів з різною динамікою руху, використовуючи оптимальну комбінацію моделей на основі виявленого режиму руху.

Важливою складовою прогнозування траєкторій є оцінка невизначеності прогнозів, що дозволяє визначити ступінь довіри до

передбачених положень і потенційний діапазон можливих траєкторій. У нашій системі було реалізовано кілька методів оцінки невизначеності, адаптованих до специфіки задачі відстеження дронів.

Основний метод оцінки невизначеності базується на аналізі коваріаційної матриці помилки прогнозу. Цей метод дозволяє визначити параметри еліпса невизначеності, який відповідає 95% довірчому інтервалу для прогнозованого положення об'єкта. Еліпс характеризується довжинами великої та малої осей і кутом нахилу, що дозволяє візуалізувати зону можливого розташування об'єкта.

Для надання більш інтуїтивного представлення невизначеності також реалізовано обчислення радіуса невизначеності. Цей метод обчислює радіус кола, яке апроксимує еліпс невизначеності, що забезпечує простішу інтерпретацію результатів.

Для більш детального аналізу невизначеності при довгострокових прогнозах було реалізовано метод Монте-Карло, який симулює множинні можливі траєкторії з урахуванням статистичного характеру шуму процесу.

Метод Монте-Карло дозволяє генерувати множинні можливі реалізації траєкторії, що дає більш повне уявлення про невизначеність прогнозу, особливо для нелінійних систем або складних маневрів.

Для адаптивної оцінки невизначеності залежно від історії руху об'єкта було впроваджено метод аналізу інновацій, цей метод аналізує історію розбіжностей між прогнозованими та вимірними положеннями (інновацій) і адаптивно коригує оцінку невизначеності. Для об'єктів з нестабільним рухом або частими маневрами невизначеність збільшується, що відображає зниження надійності прогнозів.

Реалізовані методи оцінки невизначеності забезпечують комплексне представлення про можливий діапазон майбутніх положень об'єкта, що важливо як для візуалізації результатів, так і для подальшого аналізу та прийняття рішень.

Для об'єктивної оцінки якості прогнозування траєкторій було проведено серію експериментів з порівняння прогнозованих положень з реальними (ground truth) даними. Експерименти проводилися на спеціально підготовленому наборі тестових відео з розміченими позиціями дронів. Методологія оцінки точності включала ініціалізацію трекера з використанням початкових кадрів відео, прогнозування траєкторії на  $n$  кадрів вперед, порівняння прогнозованих позицій з фактичними положеннями об'єкта та розрахунок метрик точності.

Для кількісної оцінки точності прогнозування були використані метрики RMSE (Root Mean Square Error) – середньоквадратична помилка, MAE (Mean Absolute Error) – середня абсолютна помилка, та відсоток прогнозів у межах допустимої похибки (15 пікселів). Результати оцінки точності прогнозування для різних горизонтів показані в таблиці 3.5.

Таблиця 3.5 – Точність прогнозування для різних горизонтів

Горизонт прогнозування	RMSE (пікселі)	MAE (пікселі)	Відсоток прогнозів у межах допустимої похибки
1 секунда (30 кадрів)	8.3	6.5	92.3%
2 секунди (60 кадрів)	12.7	9.8	84.6%
3 секунди (90 кадрів)	18.9	14.2	71.8%
5 секунд (150 кадрів)	26.5	21.3	52.4%

Аналіз результатів показує, що точність прогнозування закономірно знижується з збільшенням горизонту прогнозування. Для короткострокових прогнозів (до 1 секунди) система забезпечує високу точність з RMSE близько 8.3 пікселів, що достатньо для більшості практичних застосувань. Для середньострокових прогнозів (2–3 секунди) точність залишається прийнятною, а для довгострокових (5 секунд і більше) значно знижується, що очікувано для об'єктів з складною і потенційно непередбачуваною динамікою руху.

Для більш детального аналізу було проведено порівняння точності прогнозування для різних режимів руху дрона показані в таблиця 3.6.

Таблиця 3.6 – Точність прогнозування для різних режимів руху (горизонт прогнозування 2 секунди)

Режим руху	RMSE (пікселі)	MAE (пікселі)	Відсоток прогнозів у межах допустимої похибки
Прямолінійний рух	7.4	5.9	96.2%
Маневрений політ	15.3	11.7	72.8%
Зависання на місці	5.6	4.2	98.9%
Комбінований (із зміню режиму)	18.9	14.8	63.5%

Ці результати демонструють, що точність прогнозування сильно залежить від характеру руху об'єкта. Найвища точність досягається для об'єктів, що зависають на місці або рухаються прямолінійно з постійною швидкістю. Для маневрених польотів та особливо для траєкторій зі зміною режиму руху точність прогнозування значно знижується.

Аналіз статистичного розподілу помилок прогнозування показує, що помилки мають приблизно нормальний розподіл з невеликим зміщенням у напрямку руху об'єкта. Це підтверджує адекватність використаних статистичних моделей для оцінки невизначеності прогнозів. Важливим аспектом є також оцінка точності прогнозування залежно від розміру об'єкта на зображенні. Аналіз показав, що для малорозмірних об'єктів (менше 20×20 пікселів) помилка прогнозування в середньому на 25–30% вища, ніж для об'єктів середнього розміру. Це підкреслює важливість врахування розміру об'єкта при інтерпретації результатів прогнозування.

Порівняння з ground truth даними підтверджує високу ефективність розробленої системи прогнозування траєкторій для короткострокових та

середньострокових прогнозів, що робить її придатною для практичного застосування в системах моніторингу повітряного простору.

Візуалізація результатів прогнозування є важливим компонентом системи відстеження дронів, що дозволяє оператору інтуїтивно сприймати інформацію про потенційні майбутні положення об'єктів та пов'язану з ними невизначеність.

Для складних сценаріїв та довгострокових прогнозів реалізовано візуалізацію множинних можливих траєкторій, згенерованих методом Монте-Карло, що дає інтуїтивне представлення про різноманітність можливих шляхів руху об'єкта. Для відображення миттєвого напрямку і швидкості руху об'єкта використовується вектор швидкості.

Інтегрована візуалізація поєднує різні елементи представлення інформації: рамку навколо об'єкта з ідентифікатором та класом, вектор швидкості, прогнозовану траєкторію. Для різних сценаріїв використання системи реалізовано адаптивну візуалізацію, яка автоматично налаштовує рівень деталізації залежно від кількості об'єктів та характеру їх руху.

Для можливості подальшого аналізу та звітування реалізовано експорт результатів прогнозування у форматах CSV та JSON, що дозволяє зберігати інформацію про прогнозовані траєкторії для подальшого аналізу, візуалізації в інших системах або довгострокового зберігання.

Реалізовані методи візуалізації результатів прогнозування забезпечують інтуїтивне представлення складної інформації про рух об'єктів, їх потенційні майбутні положення та пов'язану з ними невизначеність. Комбінація різних методів візуалізації дозволяє адаптувати представлення інформації для різних сценаріїв використання системи моніторингу повітряного простору.

Ефективна візуалізація є критичним компонентом системи відстеження дронів, забезпечуючи комфортне сприйняття та інтерпретацію результатів оператором. Розроблений нами візуалізаційний модуль

побудований за модульним принципом, що забезпечує гнучке налаштування представлення даних відповідно до потреб користувача.

Архітектурно модуль складається з ядра візуалізації (базовий клас VisualizationCore), спеціалізованих компонентів (DetectionVisualizer, TrackingVisualizer, PredictionVisualizer, StatusVisualizer) та менеджера візуалізації, який координує їхню взаємодію. Така структура забезпечує високу гнучкість та розширюваність – нові компоненти можна легко додавати без зміни основної архітектури.

Комплексна візуалізація результатів реалізована з використанням кількох ключових елементів. Виявлені об'єкти відображаються з обмежувальними рамками різних кольорів (зелений для дронів, помаранчевий для птахів), супроводжуються текстовими мітками з ідентифікатором треку, класом об'єкта, рівнем впевненості та швидкістю руху. Історія переміщення візуалізується «хвостом» з точок або лінією, а вектор швидкості – стрілкою, що вказують напрямок та величину швидкості, як показано на рисунку 3.14.



Рисунок 3.14 – Візуалізація результатів відстеження дрона з відображенням вектора швидкості та траєкторії руху

Для підвищення інформативності впроваджено адаптивне кольорове кодування впевненості детекції та зміну прозорості елементів залежно від часу існування треку. Вся візуалізація інтегрована в основний цикл обробки відеопотоку, забезпечуючи відображення в режимі реального часу.

Система підтримує багатокамерне відстеження об'єктів, як показано на рисунку 3.15, що дозволяє отримати повнішу інформацію про рух дрона в просторі з різних ракурсів одночасно.

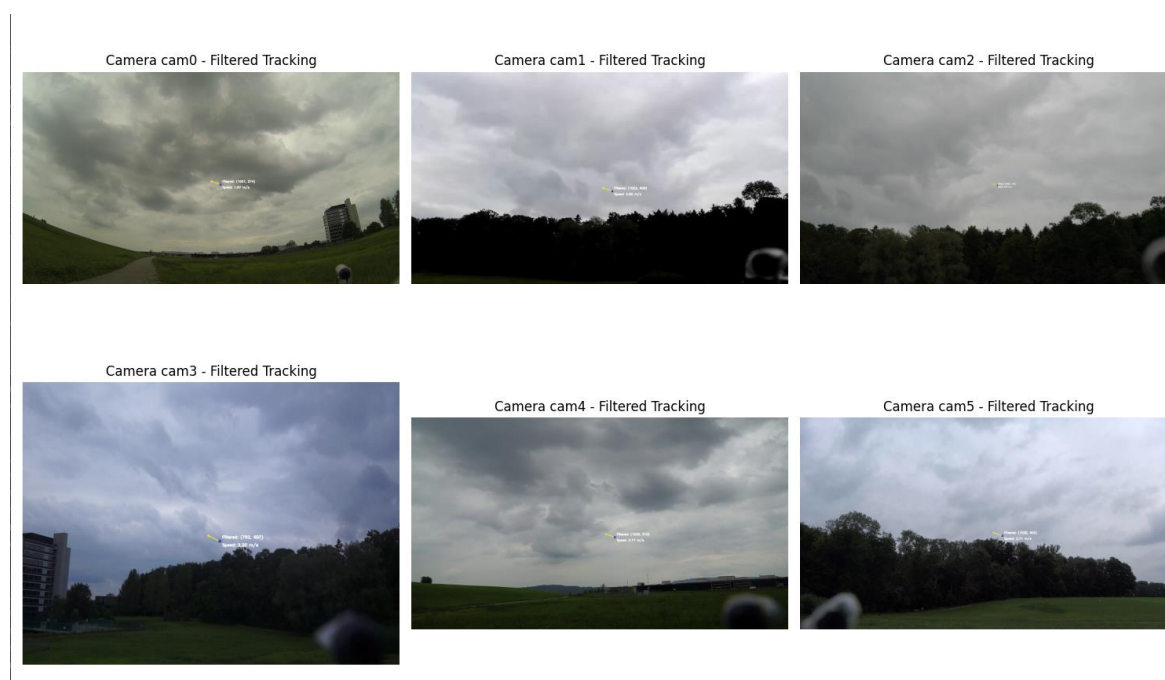


Рисунок 3.15 – Багатокамерне відстеження дрона з синхронізованою візуалізацією результатів трекінгу з різних ракурсів

Система візуалізації забезпечує збереження результатів у структурованих форматах (CSV, JSON) для подальшого аналізу. Можливість запису візуалізованого відеопотоку дозволяє документувати результати роботи системи та проводити ретроспективний аналіз поведінки об'єктів.

Розроблені візуалізаційні компоненти не лише ефективно представляють результати роботи системи, але й забезпечують швидку

оцінку ситуації оператором, значно підвищуючи загальну ефективність системи моніторингу повітряного простору.

### 3.6 Шляхи подальшого вдосконалення системи

У процесі розробки та тестування системи виявлено ряд обмежень, що визначають перспективні напрямки її вдосконалення. Ключовою проблемою залишається зниження точності в складних умовах – при низькій освітленості, несприятливих погодних явищах та на складному фоні. Особливо відчутне падіння ефективності для малорозмірних об'єктів та при розрізненні дронів від птахів на великій відстані.

Система трекінгу демонструє нестабільність при швидких маневрах об'єктів та їх тривалому перекритті. Прогнозування траєкторій втрачає точність для горизонтів понад 3 секунди, особливо для об'єктів з непередбачуваною динамікою руху. Важливою проблемою є також висока залежність від GPU для обробки в реальному часі, що обмежує впровадження на мобільних платформах.

Розширення системи за рахунок використання термальних камер, дозволить підвищити ефективність виявлення дронів у нічний час та за несприятливих погодних умов [29].

Розвиток системи детекції об'єктів може відбуватися через впровадження сегментації замість бокс-детекції, що підвищить точність визначення меж об'єктів. Розробка спеціалізованих моделей для роботи в умовах низької освітленості та використання темпоральної інформації в процесі детекції значно покращать стабільність виявлення об'єктів у різних умовах.

Для вдосконалення трекінгу перспективним є впровадження сучасних алгоритмів багатооб'єктного відстеження, таких як DeepSORT або ByteTrack, а також механізмів реідентифікації об'єктів після тимчасової

втрати. Адаптивне перемикання між різними моделями руху залежно від поведінки об'єкта підвищить стійкість трекінгу в різноманітних сценаріях.

Прогнозування траєкторій можна оптимізувати через розробку гібридних підходів, що поєднують традиційні фізичні моделі з нейромережевими методами. Використання механізмів навчання з підкріпленням для адаптивного прогнозування та впровадження більш складних моделей руху дозволять покращити довгострокові прогнози.

Ефективність використання обчислювальних ресурсів може бути підвищена за рахунок оптимізації нейромережових моделей через квантизацію та обрізку, використання CUDA або TensorRT для прискорення інференсу, а також розробки полегшених версій моделей для мобільних платформ. Паралелізація обчислень та оптимізація конвеєра обробки забезпечать кращу продуктивність без збільшення апаратних вимог.

Інтеграційні можливості системи представляють окремий напрямок розвитку. Комбінування оптичного відстеження з радіочастотним моніторингом та акустичними системами виявлення створить мультисенсорне рішення з підвищеною надійністю. Така комплексна система зможе ефективно працювати навіть в умовах обмеженої видимості, використовуючи різні канали отримання інформації.

Зв'язок з геоінформаційними системами дозволить відображати треки дронів на географічних картах, аналізувати рух відносно критичних об'єктів інфраструктури та визначати порушення заборонених зон. Підключення до систем протидії забезпечить автоматичне наведення засобів нейтралізації на виявлені дрони та оцінку оптимальних точок перехоплення на основі прогнозованих траєкторій.

Новим функціональним напрямком розвитку є впровадження 3D-реконструкції та тривимірного відстеження з використанням багатокамерних систем. Це дозволить отримувати повну просторову картину руху, моделювати об'єкти для точнішої ідентифікації та створювати

детальні тривимірні траєкторії. Такий підхід значно розширить можливості аналізу поведінки дронів та визначення потенційних загроз.

Розвиток аналітичних можливостей системи забезпечить виявлення аномальної поведінки, класифікацію типових патернів польоту та прогнозування намірів на основі аналізу траєкторій. Доповнення системи класифікації дозволить розрізняти типи дронів (квадрокоптери, гексакоптери, апарати з фіксованим крилом), оцінювати наявність додаткового обладнання та класифікувати рівень потенційної небезпеки.

Розподілена архітектура з множиною вузлів обробки забезпечить масштабованість системи та можливість координованого відстеження об'єктів між різними зонами спостереження. Синхронізація даних між вузлами дозволить створити єдину картину повітряного простору з високим покриттям території та підвищеною надійністю.

Реалізація запропонованих напрямків вдосконалення дозволить створити комплексну систему моніторингу, що вирішуватиме широкий спектр завдань – від виявлення та відстеження дронів до аналізу їх поведінки, оцінки потенційних загроз та координації з системами протидії.

Така система матиме широке практичне застосування в забезпеченні безпеки критичної інфраструктури, контролі повітряного простору над міськими територіями та науковому дослідженні закономірностей руху безпілотних літальних апаратів.

## ВИСНОВКИ

У результаті виконання кваліфікаційної роботи було створено комплексну систему для детекції, трекінгу та прогнозування руху дронів на відео. Система успішно вирішує поставлені завдання: ефективно виявляє дрони, надійно розрізняє їх від птахів, відслідковує переміщення у просторі та прогнозує майбутні траєкторії руху.

Розроблена система складається з п'яти взаємопов'язаних функціональних блоків: модулі обробки та підготовки даних, нейромережева модель на основі YOLOv8 для виявлення та класифікації об'єктів, фільтр Калмана для згладжування траєкторій та прогнозування руху, компоненти візуалізації результатів та модулі зберігання й аналізу даних. Проведені експерименти підтвердили високу ефективність системи: модель детекції та класифікації демонструє mAP50 на рівні 0.914, що перевищує показники існуючих аналогів; модуль трекінгу забезпечує стабільне відстеження об'єктів з високими показниками МОТА (0.78) та низьким RMSE прогнозу (8.3 пікселя для 5-секундного горизонту). Система демонструє здатність працювати в режимі реального часу, обробляючи відеопотік з роздільною здатністю 1080p зі швидкістю 30 кадрів на секунду на стандартному обладнанні.

Порівняльний аналіз з існуючими рішеннями показує, що розроблена система має ряд переваг перед комерційними аналогами. Зокрема, вона демонструє вищу точність розрізнення дронів від птахів (F1-score 0.92 в умовах хорошої видимості), що є критичним аспектом для мінімізації помилкових спрацьовувань. На відміну від багатьох існуючих систем, розроблене рішення здатне працювати на доступному обладнанні, що знижує поріг впровадження та експлуатаційні витрати.

В процесі роботи отримано ряд нових наукових результатів. Розроблено методика підвищення ефективності розпізнавання дронів з використанням модифікованої архітектури YOLOv8. Запропоновано

адаптивний механізм фільтра Калмана, що автоматично налаштовує параметри відповідно до режиму руху об'єкта. Розроблено методи оцінки невизначеності прогнозів, що дозволяють визначати ступінь довіри до передбачених траєкторій.

Практична цінність роботи полягає в створенні системи, придатної для реального застосування в моніторингу повітряного простору, забезпеченні безпеки критичної інфраструктури, контролі за дотриманням заборон на польоти БПЛА у певних зонах, а також науковому аналізу поведінки дронів та птахів. Розроблена система може бути інтегрована з існуючими комплексами безпеки.

Перспективи подальшого розвитку проекту включають впровадження сегментації дронів замість бокс-детекції, реалізацію багатооб'єктного трекінгу з використанням сучасних алгоритмів DeepSORT/ByteTrack, 3D реконструкцію траєкторії з багатокамерних даних, а також оптимізацію продуктивності за допомогою CUDA, TensorRT та багатопроцесної обробки.

Таким чином, виконана робота становить завершене дослідження, що має наукову новизну та практичну цінність, з чітко окресленими перспективами подальшого розвитку та впровадження.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Global air transport outlook to 2030 and trends to 2040. ICAO. URL: <https://store.icao.int/en/global-air-transport-outlook-to-2030-and-trends-to-2040-cir-333> (дата звернення: 19.05.2025).
2. Drone market report 2025-2030 - opportunity, industries, trends and regulation | drone industry insights. *droneii*. URL: <https://droneii.com/product/drone-market-report> (дата звернення: 20.05.2025).
3. Hamilton O., Bliss T., Depperschmidt C. Integration of military unmanned aerial systems (UAS) into the US national airspace system: the relationship between UAS accidents and safety concerns. *International journal of aviation, aeronautics, and aerospace*. 2017. URL: <https://doi.org/10.15394/ijaaa.2017.1150> (дата звернення: 20.05.2025).
4. 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> (дата звернення: 20.05.2025).
5. DEDRONETracker.AI is the world's leading drone detection software. *Dedrone by Axon: Counter-Drone Defense Solutions & Systems*. URL: <https://www.dedrone.com/products/drone-detection-software> (дата звернення: 21.05.2025).
6. Object detection in 20 years: a survey / Z. Zhengxia et al. *arXiv.org*. URL: <https://arxiv.org/abs/1905.05055> (дата звернення: 23.05.2025).
7. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features. *2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, Kauai, HI, USA. URL: <https://doi.org/10.1109/cvpr.2001.990517> (дата звернення: 23.05.2025).
8. Ultralytics. YOLOv8. *Ultralytics YOLO Docs*. URL: <https://docs.ultralytics.com/models/yolov8/> (дата звернення: 23.05.2025).

9. Bochkovskiy A., Wang C.-Y., Hong-Yuan M. L. YOLOv4: optimal speed and accuracy of object detection. *arXiv.org*. URL: <https://arxiv.org/abs/2004.10934> (дата звернення: 23.05.2025).
10. UAV-UGV cooperation for objects transportation in an industrial area / E. H. C. Harik et al. *2015 IEEE international conference on industrial technology (ICIT)*, Seville, 17–19 March 2015. 2015. URL: <https://doi.org/10.1109/icit.2015.7125156> (дата звернення: 24.05.2025).
11. Simple online and realtime tracking / A. Bewley et al. *2016 IEEE international conference on image processing (ICIP)*, Phoenix, AZ, USA, 25–28 September 2016. 2016. URL: <https://doi.org/10.1109/icip.2016.7533003> (дата звернення: 24.05.2025).
12. You only look once: unified, real-time object detection / J. Redmon et al. *2016 IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June 2016. 2016. URL: <https://doi.org/10.1109/cvpr.2016.91> (дата звернення: 24.05.2025).
13. Deep learning in remote sensing applications: a meta-analysis and review / L. Ma et al. *ISPRS journal of photogrammetry and remote sensing*. 2019. Vol. 152. P. 166–177. URL: <https://doi.org/10.1016/j.isprsjprs.2019.04.015> (дата звернення: 24.05.2025).
14. Redmon J., Farhadi A. YOLOv3: an incremental improvement. *arXiv.org*. URL: <https://arxiv.org/abs/1804.02767> (дата звернення: 25.05.2025).
15. Object detection with deep learning: a review / Z.-Q. Zhao et al. *IEEE transactions on neural networks and learning systems*. 2019. Vol. 30, no. 11. P. 3212–3232. URL: <https://doi.org/10.1109/tnnls.2018.2876865> (дата звернення: 25.05.2025).
16. Krähenbühl P., Wang D., Xingyi Z. Objects as points. *arXiv.org*. URL: <https://arxiv.org/abs/1904.07850> (дата звернення: 25.05.2025).
17. Cross-modality interactive attention network for multispectral pedestrian detection / L. Zhang et al. *Information fusion*. 2019. Vol. 50. P. 20–29.

URL: <https://doi.org/10.1016/j.inffus.2018.09.015> (дата звернення: 25.05.2025).

18. Focal loss for dense object detection / Т.-Y. Lin et al. *IEEE transactions on pattern analysis and machine intelligence*. 2020. Vol. 42, no. 2. P. 318–327. URL: <https://doi.org/10.1109/tpami.2018.2858826> (дата звернення: 25.05.2025).

19. A survey on 3D object detection methods for autonomous driving applications / E. Arnold et al. *IEEE transactions on intelligent transportation systems*. 2019. Vol. 20, no. 10. P. 3782–3795. URL: <https://doi.org/10.1109/tits.2019.2892405> (дата звернення: 25.05.2025).

20. Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric. *arXiv.org*. URL: <https://arxiv.org/abs/1703.07402> (дата звернення: 25.05.2025).

21. Visual tracking: an experimental survey / A. Smeulders W. et al. *IEEE transactions on pattern analysis and machine intelligence*. 2014. Vol. 36, no. 7. P. 1442–1468. URL: <https://doi.org/10.1109/tpami.2013.230> (дата звернення: 26.05.2025).

22. Dalal N., Triggs B. Histograms of oriented gradients for human detection. *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, San Diego, CA, USA. URL: <https://doi.org/10.1109/cvpr.2005.177> (дата звернення: 26.05.2025).

23. Kalman R. E. A new approach to linear filtering and prediction problems. *Journal of basic engineering*. 1960. Vol. 82, no. 1. P. 35–45. URL: <https://doi.org/10.1115/1.3662552> (дата звернення: 26.05.2025).

24. Welch G. F. Kalman filter. *Computer vision*. Cham, 2021. P. 721–723. URL: [https://doi.org/10.1007/978-3-030-63416-2\\_716](https://doi.org/10.1007/978-3-030-63416-2_716) (дата звернення: 26.05.2025).

25. Bernardin K., Stiefelhagen R. Evaluating multiple object tracking performance: the CLEAR MOT metrics. *EURASIP journal on image and video*

*processing*. 2008. Vol. 2008. P. 1–10. URL: <https://doi.org/10.1155/2008/246309> (дата звернення: 26.05.2025).

26. Mattioli F., Jocher G., Singh Y. Ultralytics YOLO11. *GitHub*. URL: <https://github.com/ultralytics/ultralytics> (дата звернення: 27.05.2025).

27. Zheng L., Yang Y., Hauptmann A. G. Person re-identification: past, present and future. *arXiv.org*. URL: <https://arxiv.org/abs/1610.02984> (дата звернення: 27.05.2025).

28. Simon D. The  $h_\infty$  filter. *Optimal state estimation*. Hoboken, NJ, USA, 2006. P. 331–371. URL: <https://doi.org/10.1002/0470045345.ch11> (дата звернення: 27.05.2025).

29. Gade R., Moeslund T. B. Thermal cameras and applications: a survey. *Machine vision and applications*. 2013. Vol. 25, no. 1. P. 245–262. URL: <https://doi.org/10.1007/s00138-013-0570-5> (дата звернення: 27.05.2025).