

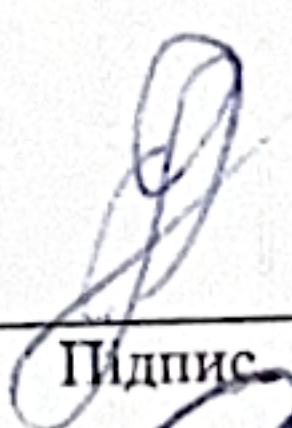
## КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

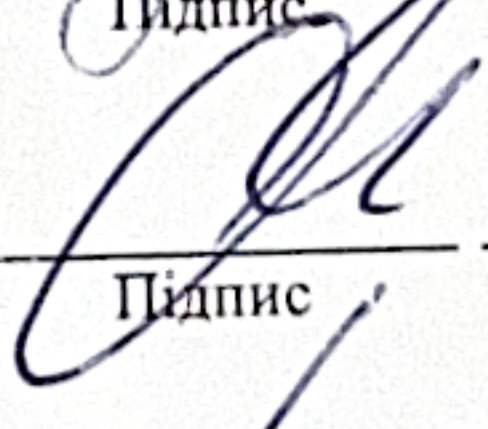
на тему Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання

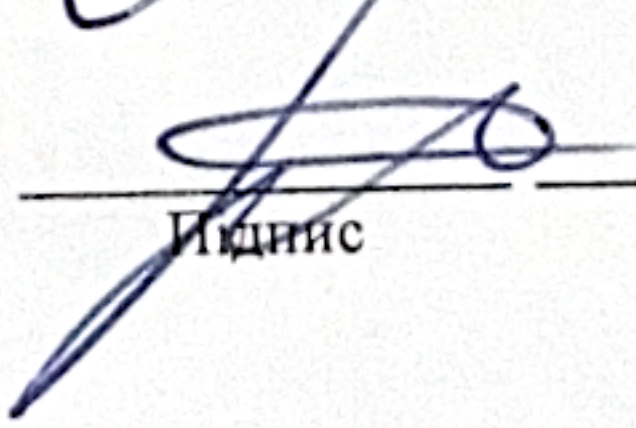
Галузь знань 12 – Інформаційні технології  
Шифр і назва галузі знань


Спеціальність 122 – Комп'ютерні науки  
Шифр і назва спеціальності

Освітня програма Комп'ютерні науки  
Назва освітньої програми

Виконав: студент групи КН-21-1  
Група виконавця  Олександр АНДРІЙЧАК  
Ім'я, ПРІЗВИЩЕ

Керівник: д.т.н., зав. кафедри КН  
Науковий ступінь, посада  Олександр БАРМАК  
Ім'я, ПРІЗВИЩЕ

Нормоконтроль: к.т.н., доц. каф. КН  
Науковий ступінь, посада  Руслан БАГРІЙ  
Ім'я, ПРІЗВИЩЕ

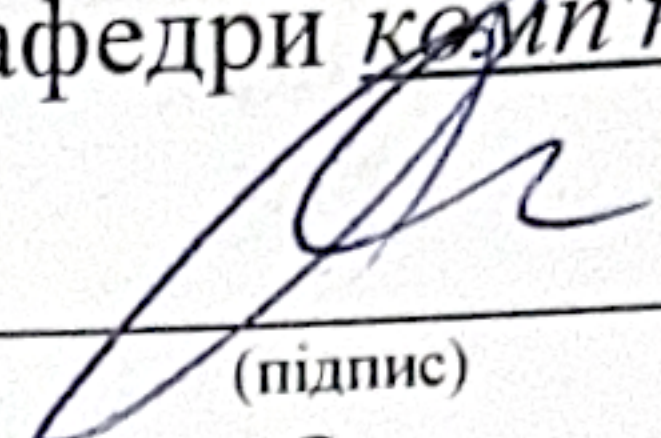
До захисту допускаю:  
зав. кафедри КН, д.т.н., професор  Олександр БАРМАК  
Ім'я, ПРІЗВИЩЕ

\_\_\_\_\_ 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
Факультет інформаційних технологій  
Кафедра комп'ютерних наук  
Освітній ступінь бакалавр  
Галузь знань 12 – Інформаційні технології  
Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук

  
(підпис)

д.т.н., професор Олександр БАРМАК

« 10 » 02 2025 року

### ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

1. Тема кваліфікаційної роботи бакалавра: «Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання»

2. Завдання видано студенту Олександр АНДРІЙЧАКУ  
(Ім'я, прізвище)

3. Керівник роботи зав. кафедри КН Олександр БАРМАК  
(посада, ім'я, прізвище)

4. Затверджено наказом університету від «07» 02 2025 р. № 23

5. Дата видачі завдання студенту: «10» 02 2025 р.

6. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи – підвищення точності виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання.

Необхідно провести аналіз методів та підходів до виявлення фортифікаційних споруд на зображенні, спроектувати метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання, виконати програмну реалізацію методу виявлення фортифікаційних споруд, провести експериментальне тестування реалізованого методу.

7. Календарний план виконання кваліфікаційної роботи бакалавра:

№	Назва етапів (розділів) кваліфікаційної роботи бакалавра	Термін виконання	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи бакалавра з керівником, складання календарного графіка виконання	січень 2025	виконано
2	Ознайомлення з предметною областю, формулювання мети і задач дослідження, визначення об'єкта та предмета дослідження	лютий 2025	виконано
3	Проектування та розроблення методу вирішення завдання, загальної архітектури програмного забезпечення, інтерфейсу користувача, вибір засобів реалізації програмного забезпечення	березень 2025	виконано
4	Створення та тестування програмного забезпечення, дослідження ефективності, висновки з виконаної роботи	квітень 2025	виконано
5	Написання пояснювальної записки, урахування зауважень керівника, оформлення згідно з вимогами	травень 2025	виконано
6	Розробка презентаційних матеріалів та попередній захист кваліфікаційної роботи	травень 2025	виконано
7	Отримання відгуку керівника, рецензії, перевірка на плагіат, нормоконтроль	червень 2025	виконано
8	Підготовка до захисту та захист кваліфікаційної роботи	червень 2025	виконано

Виконавець:

студент групи КН-21-1

Група виконавця

  
Підпис

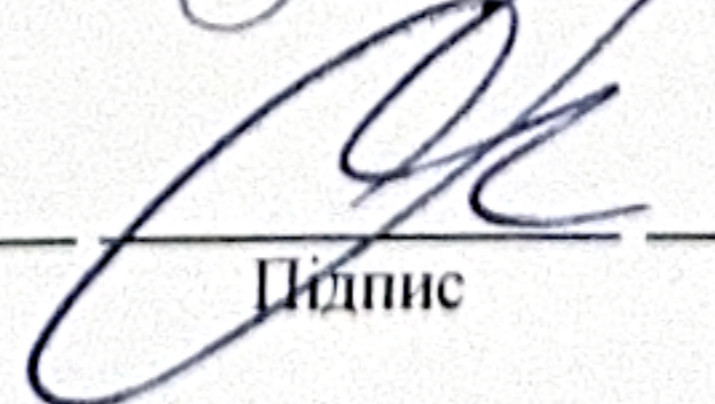
Олександр АНДРІЙЧАК

Ім'я, ПРІЗВИЩЕ

Керівник:

д.т.н., професор

Науковий ступінь, посада

  
Підпис

Олександр БАРМАК

Ім'я, ПРІЗВИЩЕ

## Анотація

Тема кваліфікаційної роботи бакалавра: «Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання»

Виконавець кваліфікаційної роботи бакалавра: студент групи КН-21-1  
Олександр АНДРІЙЧАК

Керівник кваліфікаційної роботи бакалавра: д.т.н., професор кафедри КН  
Олександр БАРМАК

Кваліфікаційна робота бакалавра містить:

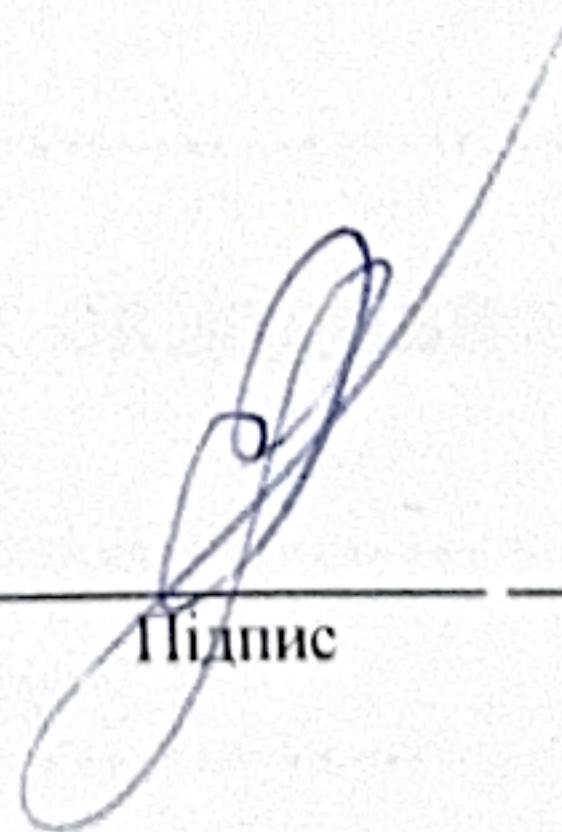
Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
61	23	3	42	2

Метою кваліфікаційної роботи бакалавра є підвищення точності виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання шляхом розробки відповідного методу.

Для досягнення зазначеної мети було проведено аналіз предметної області та огляд сучасних підходів до автоматичного виявлення об'єктів на зображеннях. Запропоновано метод виявлення фортифікаційних споруд, реалізований із використанням сучасних архітектур глибоких нейронних мереж. Наведено результати експериментальної перевірки ефективності розробленого підходу.

Ключові слова: фортифікаційні споруди, моделі глибокого навчання, зображення з БПЛА.

Виконавець: студент групи КН-21-1  
Група виконавця



Підпис

Олександр АНДРІЙЧАК  
Ім'я, ПРІЗВИЩЕ

## Зміст

Перелік скорочень .....	3
Вступ.....	4
Розділ 1 Характеристика предметної області: аналіз моделей, методів та реалізацій.....	5
1.1 Аналіз інформаційних моделей .....	5
1.2 Огляд теоретичних підходів до розв’язку подібних задач .....	7
1.3 Аналіз існуючих програмних засобів та наукових рішень .....	12
1.4 Мета, задачі та вимоги до реалізації інформаційної системи .....	14
1.5 Висновки до розділу 1 .....	14
Розділ 2 Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання .....	15
2.1 Основна ідея методу .....	15
2.2. Опис датасету для донавчання моделі глибокого навчання.....	16
2.3 Архітектура моделі глибокого навчання із сімейства YOLO .....	19
2.4. Основні кроки методу.....	23
2.5. Метрики оцінювання .....	25
2.7 Висновки до розділу 2 .....	33
Розділ 3 Експериментальне дослідження запропонованого методу.....	35
3.1 Опис застосування для експериментального дослідження .....	35
3.1.1. Бібліотеки та фреймворки для створення застосування .....	35
3.1.2. Діаграма класів застосування .....	37
3.1.3 Особливості реалізації програмних складових системи.....	38
3.2 Результати досліджень .....	44
3.3 Висновки до розділу 3 .....	53
Загальні висновки.....	55
Перелік посилань.....	57
Додатки	

### Перелік скорочень

Скорочення, термін, позначення	Пояснення
TF-IDF	Term Frequency – Inverse Document Frequency
ІНМ	Інформаційний навчальний матеріал
ІС	Інформаційна система
ІТ	Інформаційні технології
КРБ	Кваліфікаційна робота бакалавра
КН	Комп'ютерні науки
НК	Навчальний курс
НМ	Нейронна мережа
ПЗ	Пояснювальна записка
ПП	Програмний продукт
ТНМ	Тестовий навчальний матеріал
ХНУ	Хмельницький національний університет.
CLR	Common Language Runtime
BCL	Base Class Library
TF	Term Frequency

## Вступ

Кваліфікаційна робота бакалавра присвячена покращенню виявлення фортифікаційних споруд за знімками методами інтелектуального аналізу даних для оперативного ухвалення стратегічних рішень та забезпечення безпеки. Розроблена система використовує методи штучного інтелекту, зокрема глибоке навчання для автоматичного аналізу знімків з БПЛА та виявлення фортифікацій.

**Актуальність.** Розвиток та активне використання безпілотних літальних апаратів призводять до значного зростання обсягів візуальної інформації, що потребує оперативного аналізу. Зображення, отримані з БПЛА, часто містять важливі об'єкти, зокрема фортифікаційні споруди, виявлення яких має велике значення для забезпечення безпеки та ухвалення стратегічних рішень. Тому розпізнавання таких об'єктів стає важливою задачею, особливо в умовах активних бойових дій або надзвичайних ситуацій. У цьому контексті проблема підвищення точності розпізнавання фортифікаційних споруд засобами глибокого навчання набуває особливої актуальності в наш час.

**Об'єкт дослідження** – процес розпізнавання фортифікаційних споруд на зображенні з БПЛА.

**Предмет дослідження** – алгоритми, засоби та підходи глибокого навчання до виявлення об'єктів на зображенні.

**Мета кваліфікаційної роботи бакалавра** – підвищення точності виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання.

**Завдання кваліфікаційної роботи бакалавра.** Провести аналіз предметної області виявлення фортифікаційних споруд на зображеннях з БПЛА. Розглянути методи виявлення об'єктів на зображеннях за допомогою глибокого навчання та особливості їх застосування для обробки БПЛА-зображень. Проаналізувати наявні рішення щодо виявлення фортифікаційних споруд. Розробити та дослідити точність методу виявлення фортифікаційних споруд на зображеннях з БПЛА із використанням глибокого навчання.

## **Розділ 1 Характеристика предметної області: аналіз моделей, методів та реалізацій**

### **1.1 Аналіз інформаційних моделей**

Розвиток військових технологій та широке застосування безпілотних літальних апаратів (БПЛА) в Україні, особливо в умовах активних бойових дій, обумовлюють необхідність постійного вдосконалення методів автоматизованого розпізнавання об'єктів на зображеннях з БПЛА [1]. Одним із ключових завдань сучасної розвідки є виявлення фортифікаційних споруд – укріплень, окопів, бліндажів та інших оборонних конструкцій – що має стратегічне значення для ефективного планування оборонних операцій, забезпечення безпеки підрозділів та аналізу оперативної обстановки на місцевості [2].

Процес виявлення фортифікаційних споруд передбачає побудову відповідної інформаційної моделі об'єкта, яка дозволяє відокремити його серед різноманітних елементів природного чи техногенного середовища. Така модель повинна точно відображати характерні ознаки фортифікацій: геометричну форму, розміри, текстуру поверхні, характер контурів, типові тіньові ефекти, а також специфічні спектральні характеристики, які можуть відрізнити їх від навколишньої території.

Ефективна побудова інформаційних моделей вимагає врахування низки викликів, пов'язаних зі специфікою зображень з БПЛА, а саме [3]:

- змінність масштабу через різну висоту польоту апаратів;
- варіації освітлення залежно від часу доби та погодних умов;
- наявність перешкод, таких як рослинність, сніг, руйнування інфраструктури;
- можливість навмисного маскування об'єктів супротивником.

Основні підходи до побудови інформаційних моделей виявлення об'єктів включають [4]:

- моделі ознак – формування векторів ключових ознак, таких як контури, кути, локальні текстури, для подальшої класифікації або локалізації об'єктів;

- геометричні моделі – використання шаблонів або параметричних описів типових форм споруд для виявлення об'єктів за їх структурними характеристиками (наприклад, форма окопів, бліндажів, оборонних ровів);

- піксельні моделі (моделі семантичної або інстанс-сегментації) – побудова точних масок, які відокремлюють об'єкти від фону на рівні кожного пікселя.

З появою та розвитком методів глибокого навчання особливу популярність набули згорткові нейронні мережі (CNN) [5]. Ці моделі мають здатність автоматично виявляти значущі ознаки об'єктів без необхідності ручного формування векторів ознак. Багаторівневе представлення даних у глибоких мережах дозволяє успішно розпізнавати навіть частково замасковані, пошкоджені або деформовані фортифікаційні споруди.

Особливе значення у даному контексті мають такі типи моделей глибокого навчання:

- моделі класифікації (наприклад, ResNet [6], EfficientNet [7] – для визначення наявності/відсутності фортифікацій на зображенні або в його фрагменті;

- моделі об'єктного детектування (YOLO [8], Faster R-CNN [9]) – для локалізації об'єктів за допомогою обмежувальних прямокутників;

- моделі сегментації (U-Net [10], DeepLab [11])– для точного виділення форми фортифікацій у піксельному форматі.

Для ефективного навчання таких моделей критично важливою є наявність якісно розмічених даних, які містять:

- класифікаційні мітки (наявність або відсутність фортифікацій);

- координати обмежувальних прямокутників (bounding boxes)

або детальні піксельні маски для завдань семантичної чи інстанс-сегментації.

У бойових умовах, де якість знімків може бути нерівномірною через швидкі зміни обстановки, вибір відповідних архітектур моделей та механізмів підвищення стійкості до шумів і артефактів стає надзвичайно важливим. Серед

актуальних підходів варто виділити використання попередньо навченої моделі на великій кількості загальних зображень (transfer learning) [12], застосування технік аугментації даних (обертання, масштабування, зміна яскравості, додавання шуму), розробку моделей зі зменшеною вимогою до обсягу навчальних даних (наприклад, few-shot learning [13]).

Таким чином, аналіз існуючих інформаційних моделей виявлення фортифікаційних споруд свідчить про високу ефективність сучасних методів глибокого навчання, у задачах автоматизованої обробки знімків з БПЛА.

## **1.2 Огляд теоретичних підходів до розв'язку подібних задач**

Виявлення об'єктів на зображеннях, отриманих з безпілотних літальних апаратів (БПЛА), є однією з ключових задач комп'ютерного зору, яка відіграє важливу роль у військовій розвідці, моніторингу територій, пошуково-рятувальних операціях тощо [14]. Мета цієї задачі полягає у локалізації об'єктів на зображенні шляхом визначення обмежувальних рамок (bounding boxes) [15] та класифікації кожного знайденого об'єкта відповідно до певної категорії.

У процесі розвитку комп'ютерного зору було запропоновано декілька принципово різних теоретичних підходів до вирішення цієї задачі. Вони умовно поділяються на три основні групи: двоетапні методи, одноетапні методи та гібридні архітектури на базі трансформерів [16].

### **Двоетапні підходи**

Першим суттєвим проривом у виявленні об'єктів стали так звані двоетапні методи. Ключова ідея двоетапних методів полягає у розділенні процесу виявлення на дві послідовні фази, як це наочно продемонстровано на рисунку 1.1.

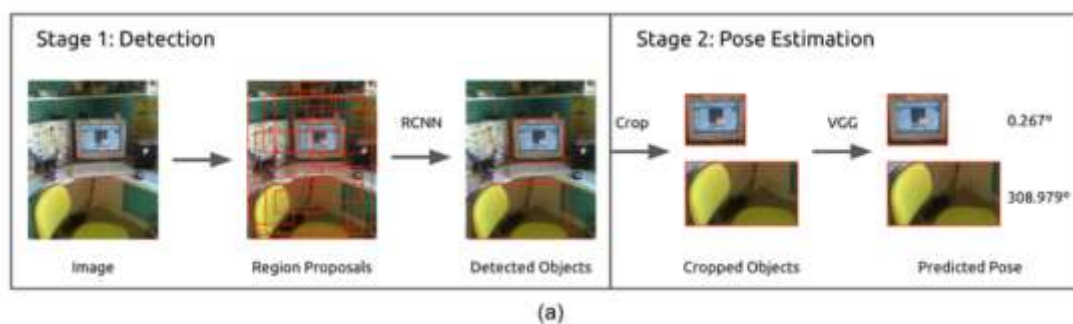


Рисунок 1.1 – Двоетапний метод виявлення об'єктів. [16]

На першому етапі відбувається генерація регіонів-пропозицій (Region Proposals) [17] – визначаються потенційні ділянки на зображенні, які можуть містити об'єкти. Як видно на рисунку, ці ділянки представлені набором червоних рамок, що охоплюють різні частини зображення. На другому етапі для кожної з цих виділених ділянок проводиться класифікація та уточнення – визначається клас об'єкта, що знаходиться в межах рамки, та коригуються координати обмежувальної рамки для більш точної локалізації.

Першою моделлю, що заклала основу цього підходу, стала R-CNN (Regions with CNN features). Вона використовувала зовнішній алгоритм (селективний пошук) для генерації кількох тисяч кандидатних регіонів, кожен із яких оброблявся окремою згортковою мережею для отримання ознак та подальшої класифікації.

На першому етапі зображення проходить через послідовні згорткові шари з функціями активації (зазвичай ReLU) та операціями субдискретизації (pooling), що дозволяє виявляти ключові просторові ознаки: краї, текстури, форми. Після цього відбувається згортання просторових ознак у вектор ознак, який надходить на повнозв'язні шари (fully connected layers), де відбувається класифікація з використанням, зокрема, SoftMax-функції для отримання ймовірнісного розподілу по класах.

Подальші вдосконалення привели до появи моделей Fast R-CNN і Faster R-CNN. Fast R-CNN інтегрувала обчислення ознак для всього зображення одразу, що значно пришвидшило процес. Faster R-CNN запропонувала замість зовнішнього механізму селективного пошуку власну підмережу для генерації

регіонів-пропозицій (Region Proposal Network, RPN), що ще більше оптимізувало обчислення [18].

У завданнях виявлення фортифікаційних споруд на зображеннях з БПЛА CNN є особливо корисними, оскільки дозволяють вловлювати складні просторові патерни, характерні для траншей, бліндажів чи окопів, навіть якщо вони частково масковані або мають незначний контраст із фоном. Проте базових CNN-архітектур недостатньо для повноцінної детекції з локалізацією, оскільки вони, як правило, працюють у форматі класифікації "все зображення – один клас".

### Одноетапні підходи

На противагу двоетапним методам, одноетапні підходи (рисунок 1.2) орієнтовані на швидкість та спрощення архітектури. Вони безпосередньо перетворюють задачу виявлення об'єктів на задачу регресії та класифікації для кожної ділянки зображення без проміжної фази генерації регіонів.

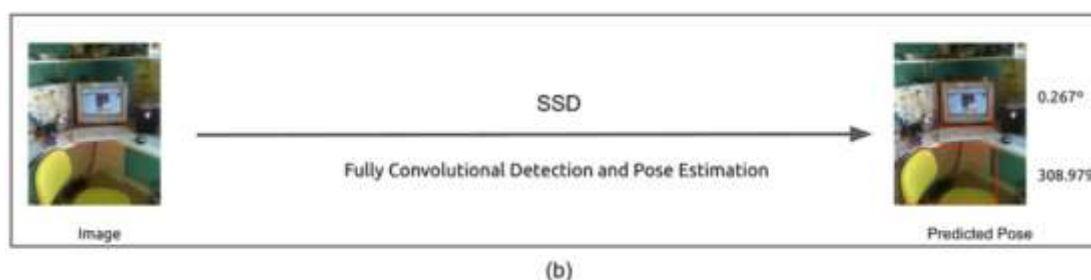


Рисунок 1.2 – Одноетапний метод виявлення об'єктів. [16]

Одним із перших ефективних рішень такого типу став SSD (Single Shot Multibox Detector) [19]. SSD пропонував прогнозування обмежувальних рамок і класів для великої кількості попередньо визначених шаблонів (анкерних коробок) у різних масштабах, що дозволяло краще працювати з об'єктами різного розміру.

Ще більш суттєвий вплив на розвиток цієї галузі мала модель YOLO (You Only Look Once). Її ключова ідея полягала у тому, що все зображення розбивається на сітку, і кожна комірка сітки відповідає за передбачення об'єктів, які розташовані всередині неї.

Перша версія YOLO мала деякі обмеження у точності локалізації, особливо для дрібних об'єктів, проте показала безпрецедентну швидкість обробки. Надалі у

версіях YOLOv2, YOLOv3 та YOLOv4 було значно покращено як точність, так і стійкість моделей завдяки використанню таких технік, як [20]:

- введення анкер-боксів;
- використання багаторівневих ознак (feature pyramid);
- поліпшення функцій втрат для кращої локалізації.

Останні версії YOLO (зокрема YOLOv5, YOLOv6, YOLOv7, YOLOv8 [21]) продовжують удосконалення архітектури, зокрема через впровадження глибших і легших моделей, використання новітніх підходів до оптимізації обчислень (наприклад, автоматичне зміщення анкерів, Mosaic-авгментація даних, використання attention-механізмів) та забезпечення високої гнучкості для завдань різної складності.

Переваги одноетапних підходів, і зокрема YOLO, полягають у:

- надзвичайно високій швидкості обробки (до сотень кадрів за секунду);
- можливості використання на обмежених за ресурсами пристроях;
- простоті налаштування та адаптації до нових задач.

На відміну від двоетапних моделей, таких як Faster R-CNN, YOLO виконує детекцію у режимі реального часу, що критично важливо в умовах обробки даних з БПЛА чи супутникових потоків. Завдяки щільній згортковій архітектурі та поділу зображення на сітку з подальшим прогнозуванням меж та класів об'єктів, YOLO демонструє високу продуктивність навіть на ресурсно обмежених системах. Це робить модель не лише доцільною, а й стратегічно виправданою для завдань військової розвідки або оперативного моніторингу оборонних ліній. Завдяки своїй здатності одночасно прогнозувати координати обмежувальних рамок і відповідні класи об'єктів у межах одного проходу, YOLO виявляє навіть невеликі або частково перекриті цілі на складному фоні – як, наприклад, на зображенні 1.3.

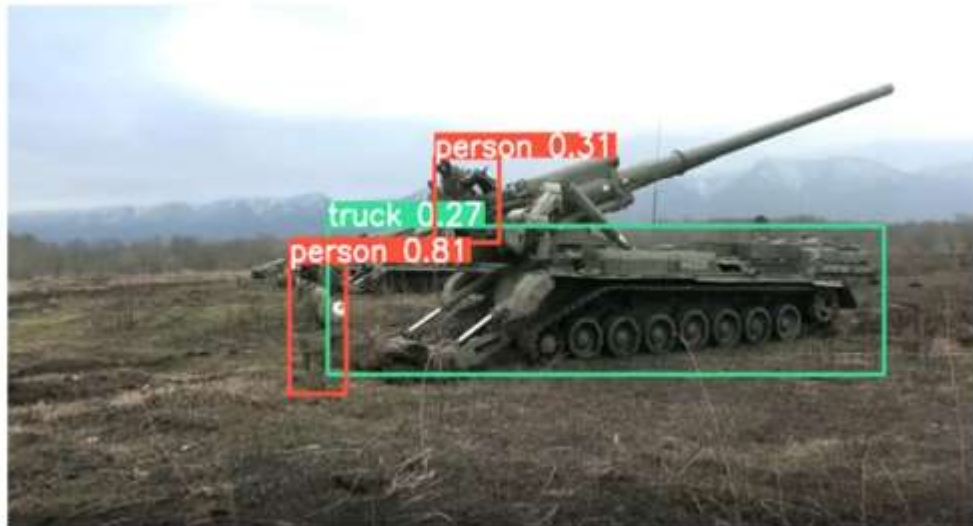


Рисунок 1.3 – Результати розпізнавання та ідентифікації двох осіб та вантажівки у відеоряді YOLO [3]

Модель забезпечує збалансоване поєднання точності та швидкодії, що особливо цінне при обробці відеопотоків або серійних аерофотознімків, де важлива не лише якість, а й оперативність аналізу. Як зазначено у джерелі [3], YOLO є придатним інструментом для використання в мобільних і розподілених системах військового або цивільного моніторингу, де класичні детектори виявляються надто повільними або ресурсоємними.

Для задачі виявлення фортифікаційних споруд, які часто мають геометрично впізнавані форми та розташовані в умовах складного фону, така модель дозволяє досягати високої точності за умови правильної підготовки тренувальної вибірки.

### **Архітектури на базі трансформерів**

Новим етапом у розвитку виявлення об'єктів стало впровадження архітектур, побудованих на базі трансформерів. Однією з перших таких моделей стала DETR (DEtection TRansformer) [22], що запропонувала уніфікований підхід до виявлення об'єктів як задачу прямого передбачення набору об'єктів без необхідності генерації кандидатних регіонів або використання анкерів.

DETR застосовує механізм самоуваги (self-attention) [23], що дозволяє моделі краще враховувати глобальні взаємозв'язки між об'єктами на зображенні. Однак основним недоліком таких моделей наразі залишається значна

обчислювальна вартість тренування та інференсу, що ускладнює їх застосування у задачах реального часу.

Отже, сучасні підходи до задачі виявлення об'єктів на зображеннях з БПЛА розвиваються у трьох основних напрямках [24]:

- двоетапні методи орієнтовані на максимальну точність і зазвичай застосовуються там, де допустима вища обчислювальна складність, що може бути корисним у детальному аналізі фортифікаційних споруд;

- одноетапні методи, зокрема сімейство YOLO, є оптимальними для задач, які потребують високої швидкості обробки у поєднанні із якісною локалізацією об'єктів, що робить YOLO ефективним рішенням для автоматизованого виявлення фортифікаційних споруд на аерофотознімках;

- трансформерні архітектури відкривають нові можливості завдяки гнучкому врахуванню глобальних взаємозв'язків на зображенні, проте наразі потребують удосконалення з погляду ефективності, щоб їх можна було застосовувати в аналізі великих наборів зображень з БПЛА.

### **1.3 Аналіз існуючих програмних засобів та наукових рішень**

Розглянемо наявні програмні рішення та наукові дослідження, які вже сьогодні застосовуються для виявлення об'єктів на аерофотознімках. Ці рішення базуються на використанні можливостей штучного інтелекту, зокрема методів глибокого навчання. Завдяки сучасним технологічним досягненням існує можливість ефективно ідентифікувати об'єкти різного типу на зображеннях, отриманих із безпілотних літальних апаратів (БПЛА), що є критично важливим завданням для військової справи, моніторингу територій та безпеки.

З метою оцінки поточного рівня розвитку відповідних технологій розглянемо сучасні системи і підходи, які застосовуються для вирішення подібних задач, їхні характеристики, методи та особливості. Застосування ШІ для аналізу аерофотознімків є доволі поширеною практикою, особливо із зростанням обчислювальних потужностей та розвитку комп'ютерного зору [25].

YOLO (You Only Look Once) [26] – це одна з найпопулярніших архітектур нейронних мереж для задач виявлення об'єктів. YOLO дозволяє в режимі реального часу здійснювати локалізацію та класифікацію об'єктів на зображеннях, у тому числі на даних з БПЛА. Основною перевагою YOLO є її висока швидкість роботи та точність. Вона розглядає всю картинку одразу, на відміну від багатьох інших моделей, що аналізують її по частинах. Це забезпечує ефективне виявлення навіть невеликих об'єктів, що дуже важливо для виявлення фортифікаційних споруд на великих територіях. Однак, недоліком може бути зниження точності при виявленні дуже дрібних або частково прихованих об'єктів.

Mask R-CNN [5] – це ще один відомий метод для сегментації та виявлення об'єктів на зображеннях. На відміну від YOLO, Mask R-CNN не тільки визначає межі об'єкта, а й створює його точну маску, що робить його дуже придатним для завдань високоточної ідентифікації, таких як виявлення складних або частково замаскованих фортифікаційних споруд.

Перевагами Mask R-CNN є висока точність локалізації та можливість обробки складних сцен. Основний недолік – велике споживання обчислювальних ресурсів та більш повільна обробка порівняно з іншими моделями.

Detectron2 [27] – це платформа від Facebook AI Research для побудови і тренування моделей комп'ютерного зору [18]. Вона підтримує як виявлення об'єктів, так і їхню сегментацію. Detectron2 активно використовується в задачах аналізу зображень з БПЛА, зокрема в оборонній сфері для виявлення укріплень та техніки. Серед переваг Detectron2 можна виділити модульність, підтримку новітніх алгоритмів і легкість налаштування під конкретну задачу.

У рамках виконання роботи передбачається використання сучасних алгоритмів комп'ютерного зору для виявлення об'єктів на зображеннях, отриманих з безпілотних літальних апаратів (БПЛА). Цей тип даних вирізняється високою роздільною здатністю, широким охопленням територій та змінними умовами зйомки, що висуває додаткові вимоги до вибору моделей.

Таким чином, з огляду на специфіку джерела даних, а саме знімків з БПЛА, обрана методологія спрямована на досягнення балансу між точністю та

продуктивністю, що є критично важливим для ефективного аналізу та подальшого використання результатів у прикладних задачах моніторингу або розвідки.

#### **1.4 Мета, задачі та вимоги до реалізації інформаційної системи**

Отже, в результаті наведеного аналізу, метою роботи є підвищення точності виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання. Для досягнення поставленої мети сформовані наступні завдання:

- проаналізувати методи та підходи до виявлення фортифікаційних споруд на зображеннях з БПЛА засобами глибокого навчання;
- розробити метод виявлення фортифікаційних споруд на аерофотознімках за допомогою технологій глибокого навчання;
- реалізувати запропонований метод;
- провести тестування розробленого методу на реальних зображеннях з БПЛА та оцінити точність виявлення фортифікаційних споруд.

#### **1.5 Висновки до розділу 1**

Аналіз сучасних наукових підходів і програмних рішень у сфері виявлення об'єктів на зображеннях засвідчив, що методи глибокого навчання, зокрема YOLO, Mask R-CNN та Detectron2, демонструють високу ефективність у задачах розпізнавання та сегментації. Це обґрунтовує їх доцільність для виявлення фортифікаційних споруд на зображеннях з БПЛА. Водночас ефективне застосування зазначених технологій потребує адаптації до специфіки цільових об'єктів і характеристик вхідних даних. У результаті аналізу підтверджено актуальність дослідження, потребу подальших розвідок і перспективи розвитку систем автоматизованого виявлення.

## **Розділ 2 Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання**

### **2.1 Основна ідея методу**

У роботі запропоновано, для розробки методу використати сучасну модель глибокого навчання із сімейства YOLOv8 (You Only Look Once version 8), яка призначена для одноетапного виявлення об'єктів на зображеннях у режимі реального часу. Основна ідея полягає у тому, щоб адаптувати цю модель для задачі автоматизованого виявлення фортифікаційних споруд на знімках, отриманих із безпілотних літальних апаратів (БПЛА).

Фортифікаційні споруди, як правило, мають чітко окреслену геометричну структуру, але водночас можуть бути частково замасковані, розташовані на складному природному або урбаністичному фоні, а також мати значну варіативність у розмірах і формах. Ці особливості висувають вимоги до точності, стійкості до шуму та здатності моделі до узагальнення в умовах обмеженої кількості навчальних прикладів.

YOLOv8 є покращеною версією попередніх моделей серії YOLO та поєднує в собі високу швидкість, компактність архітектури й конкурентну точність [28]. На відміну від двоетапних моделей, таких як Faster R-CNN, YOLOv8 виконує виявлення об'єктів за один прохід через мережу, що дозволяє значно зменшити затрати часу на обробку і зробити систему придатною для використання у практичних умовах (наприклад, у польових розгортаннях або на борту БПЛА).

У межах реалізації методу планується використання попередньо натренованої моделі YOLOv8, яка проходить етап донавчання (fine-tuning) на спеціалізованому датасеті, що містить зображення із прикладами фортифікаційних споруд різного типу. Fine-tuning дозволяє адаптувати модель до особливостей цільової вибірки: масштабу об'єктів, специфіки зйомки (кут нахилу, освітлення, перешкоди), а також структури фону.

Навчання відбувається на анотаціях, що вказують координати bounding box-ів навколо цільових об'єктів, а також відповідні їм класи. Виявлені об'єкти можуть бути як ізольованими, так і частинами більших оборонних систем. У процесі обробки модель прогнозує місцеположення та розміри об'єктів, формуючи обмежувальні рамки (bounding boxes) навколо потенційних цілей.

Завдяки високій швидкості роботи YOLOv8 забезпечує можливість використання в реальному часі, що є критично важливим у контексті моніторингу з БПЛА та оперативного прийняття рішень.

У підсумку, використання YOLOv8 у комбінації з цільовим донавчанням на спеціалізованому датасеті забезпечує ефективне, масштабоване та технічно доцільне рішення для задачі виявлення оборонних об'єктів.

## **2.2. Опис датасету для донавчання моделі глибокого навчання**

Для реалізації методу виявлення фортифікаційних споруд на основі зображень з безпілотних літальних апаратів було використано датасет Kalumbia\_map (рис. 2.1), розміщений на платформі Roboflow [29]. Цей набір даних містить понад 500 зображень, на яких вручну розмічено об'єкти фортифікаційного призначення, зокрема окопи, бліндажі, траншеї тощо.

Платформа Roboflow [30] забезпечує зручне середовище для завантаження, обробки та розмітки зображень. Розмітка виконується шляхом візуального виділення об'єктів на зображеннях – як правило, у вигляді прямокутників (bounding boxes), що дозволяє створити підготовлені вибірки для подальшого навчання моделей комп'ютерного зору. У процесі підготовки використовувалися можливості базової розмітки без застосування сегментації, що дає змогу зосередитися на загальному виявленні об'єктів на зображеннях, без потреби у високоточних координатах або поділі на підкласи.

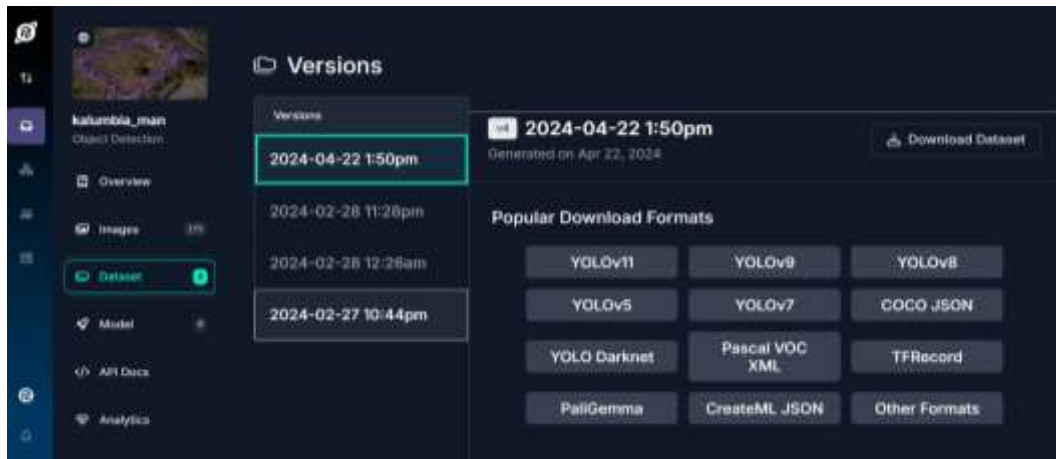


Рисунок 2.1 – Вигляд датасету на платформі Roboflow[30]

У процесі підготовки вхідних даних було здійснено аналіз наявного датасету з метою забезпечення його релевантності поставленому завданню обробки зображень. У результаті було сформовано підмножину, що включає понад 500 високоякісних зразків. Кожне зображення супроводжується анотацією у вигляді точно визначених координат областей, в яких присутні фортифікації. Ці розмічені області слугують навчальними прикладами для моделі YOLOv8, що виконує задачу детекції фортифікаційних споруд на основі зображень.

Для забезпечення методологічно коректного навчання моделі датасет було автоматизовано розподілено на дві підмножини: тренувальну (Train) та валідаційну (Validation), відповідно до стандартних пропорцій. Такий поділ дозволяє реалізувати повний цикл навчання: від оптимізації параметрів моделі на тренувальній вибірці до моніторингу її узагальнювальної здатності на валідаційних прикладах. Застосування зазначеної стратегії сприяє підвищенню достовірності оцінювання результатів і мінімізації ризику перенавчання, що є особливо важливим у контексті використання моделі для аналізу зображень, отриманих із камер безпілотних літальних апаратів.

На рисунку 2.2 наведено приклад зображення з датасету з відповідною розміткою.

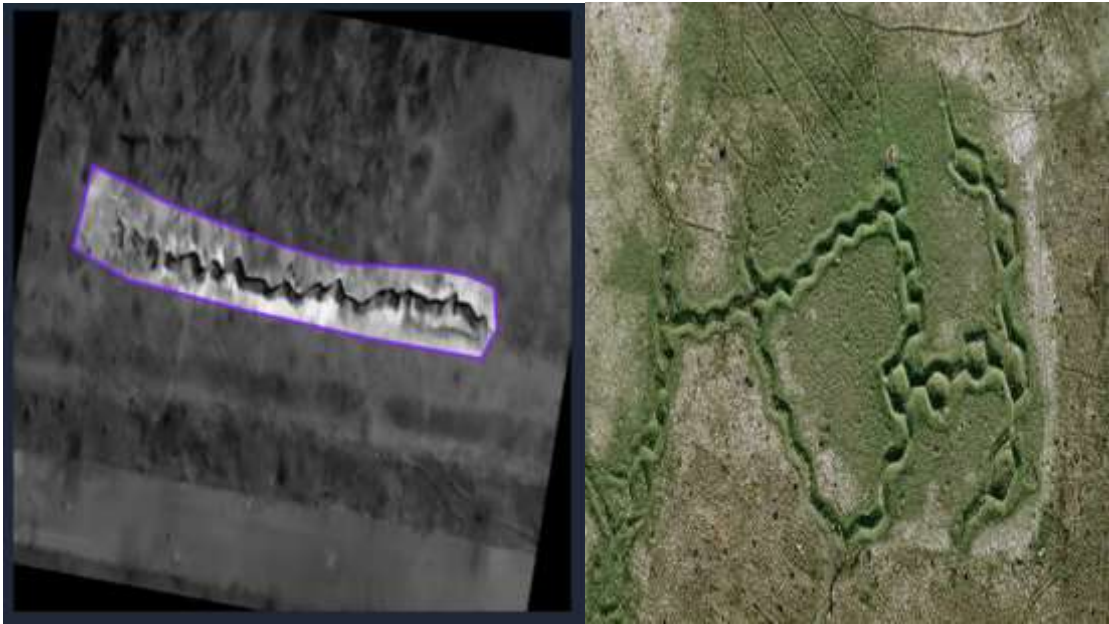


Рисунок 2.2 – Приклади зображень фортифікаційних споруд з БПЛА[29]

Таким чином, у якості вхідних робочих даних для навчання та валідації нейромережевої моделі було використано адаптований датасет *Kalumbia\_man*, що містить набір високоякісних зображень із детальною розміткою фортифікаційних споруд. Цей датасет було спеціально підготовлено з урахуванням вимог, що висуваються до структурованих даних при використанні моделі YOLOv8. Зокрема, зображення були приведені до єдиного формату, а відповідні анотації адаптовано до структури, яка включає координати обмежувальних рамок та відповідні класи об'єктів, що необхідні для коректного функціонування алгоритмів детекції.

Під час адаптації датасету було враховано особливості функціонування YOLOv8, зокрема її здатність ефективно працювати із зображеннями різного розміру та складності. Це дозволило забезпечити відповідність вхідних даних як структурно, так і змістовно: всі зразки мають точну прив'язку до об'єктів інтересу, що дає змогу моделі навчатися на реалістичних сценаріях, подібних до тих, що можуть бути зафіксовані камерами безпілотних літальних апаратів.

Використання *Kalumbia\_man* як основи для навчання моделі дозволяє не лише зменшити обсяг попередньої обробки, а й забезпечити стабільність і якість результатів, завдяки відповідності структури датасету сучасним стандартам. Це

робить його ефективним інструментом для вирішення задач виявлення фортифікаційних споруд у складних польових умовах.

### 2.3 Архітектура моделі глибокого навчання із сімейства YOLO

Для реалізації задачі виявлення фортифікаційних споруд на зображеннях, отриманих із безпілотних літальних апаратів (БПЛА), використовується сучасна архітектура глибокої нейронної мережі YOLOv8 (You Only Look Once, Version 8). Ця модель була обрана завдяки її здатності забезпечувати точне виявлення об'єктів у реальному часі, що є критично важливим для обробки великих обсягів даних від БПЛА. Дана архітектура є послідовним продовженням попередніх поколінь моделей YOLO та включає численні удосконалення, спрямовані на підвищення продуктивності, точності та ефективності обчислень. На рисунку 2.3 зображено структурну архітектуру моделі YOLO, яка реалізує підхід до одночасного виявлення, локалізації та класифікації об'єктів на зображенні в рамках єдиної нейронної мережі [31].

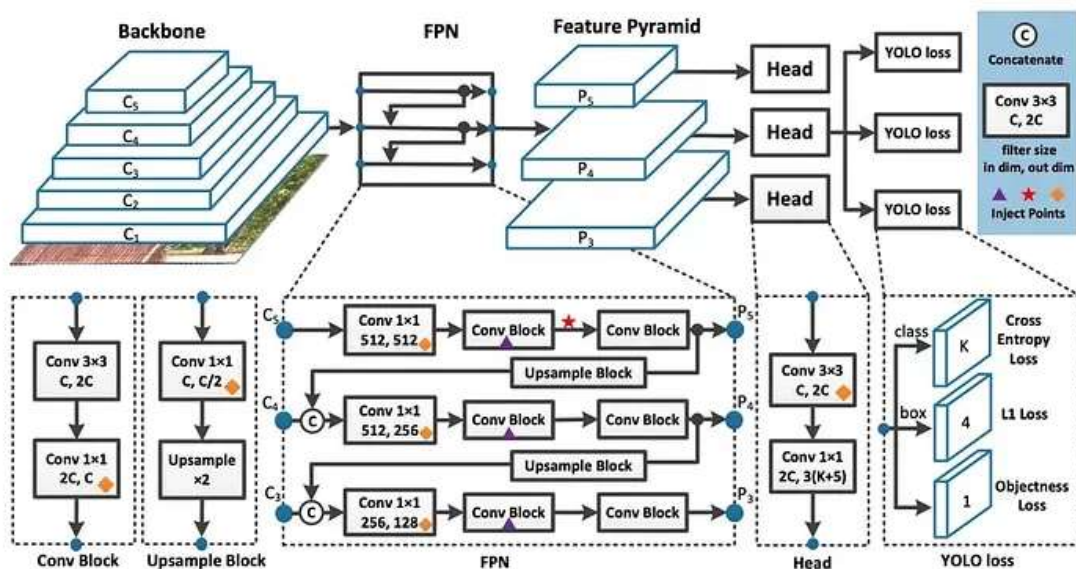


Рисунок 2.3 – Архітектура YOLOv8[31]

Архітектура моделі YOLO (You Only Look Once) складається з кількох ключових компонентів, кожен з яких виконує специфічну функцію в процесі

аналізу зображення та виявлення об'єктів. Основні складові цієї архітектури включають мережу вилучення ознак (Backbone), мережу піраміди ознак (Feature Pyramid Network, FPN), блок передбачення (Head), а також модуль функції втрат (YOLO Loss). Сукупне функціонування цих блоків забезпечує можливість моделі ефективно опрацьовувати зображення в реальному часі, досягаючи високої швидкості та точності виявлення об'єктів навіть на складних або гетерогенних вхідних даних.

Backbone є базовою частиною архітектури, що відповідає за початкове вилучення візуальних ознак з вхідного зображення. Він побудований як послідовність згорткових шарів, які дозволяють моделі поступово зменшувати просторову роздільну здатність вхідного зображення, водночас концентруючи найважливішу інформацію, яка характеризує об'єкти на зображенні. Рівні ознак, позначені як C1–C5, відповідають різним етапам цієї обробки: від базових текстур і країв (нижчі рівні) до складних семантичних структур, таких як форми, контури об'єктів і навіть контекст сцени (вищі рівні). Саме завдяки цьому модуль Backbone забезпечує потужне представлення зображення, що лягає в основу подальшого аналізу [31].

Мережа піраміди ознак (FPN) виконує критично важливу роль у багаторівневому поєднанні ознак, вилучених на різних етапах обробки. Застосовуючи механізми апсемплінгу (тобто збільшення просторового розміру ознак) та з'єднання відповідних шарів через пропущені зв'язки (skip connections), FPN дозволяє поєднати детальну локальну інформацію з високорівневою семантикою. У результаті формується ієрархічне представлення ознак, здатне однаково добре виявляти як малі об'єкти (наприклад, невеликі фрагменти траншей), так і великі структури (сукупності фортифікацій або топографічні контексти). Такий багатомасштабний підхід дозволяє зберігати баланс між точністю локалізації та семантичною повнотою.

Блок передбачення (Head) обробляє кожен рівень сформованої ознакової піраміди (позначені як P3, P4, P5) окремо. На цьому етапі генеруються три основні типи прогнозів: координати прямокутника, що окреслює об'єкт (bounding box),

оцінка наявності об'єкта у визначеному регіоні (objectness score), а також вектор імовірностей для кожного можливого класу об'єктів. Кожен масштаб ознак дозволяє ідентифікувати об'єкти певного розміру, тож одночасна робота з кількома масштабами забезпечує ефективне покриття всієї сцени. Така архітектура є надзвичайно ефективною для практичних застосувань, зокрема в контексті аерофотознімків з безпілотних літальних апаратів, де спостерігається велике різноманіття об'єктів за розміром, формою та розташуванням.

Фінальний елемент архітектури – YOLO Loss – є спеціалізованим модулем обчислення функції втрат, що забезпечує навчання моделі шляхом оцінювання якості передбачень. Цей модуль складається з кількох компонентів: функції втрат локалізації, яка оцінює відхилення передбачених координат об'єкта від реальних; функції втрат об'єктності, що визначає ступінь упевненості моделі у наявності об'єкта в певній області; а також функції втрат класифікації, яка оцінює правильність визначення класової належності. Усі ці компоненти зважуються відповідно до важливості кожного завдання та об'єднуються в єдину мету оптимізації, що забезпечує збалансоване навчання моделі.

У поєднанні всі ці компоненти формують архітектуру, яка дозволяє моделі YOLO здійснювати високошвидкісне виявлення об'єктів, зберігаючи при цьому прийнятний рівень точності. Завдяки компактності, модульності та ефективності кожного з блоків, ця архітектура є ідеальною для використання на пристроях з обмеженими обчислювальними ресурсами. У практичному застосуванні, зокрема для виявлення фортифікаційних споруд, траншей або інших військових об'єктів на зображеннях з дронів, модель демонструє стабільну продуктивність, здатну забезпечити реальний внесок у вирішення прикладних задач.

Загалом, архітектура YOLO відзначається не лише інженерною ефективністю, але й практичною придатністю до широкого спектру задач комп'ютерного зору. Високий рівень адаптивності до різних масштабів, здатність до навчання на гетерогенних датасетах та баланс між швидкістю та точністю роблять її оптимальним вибором для застосування в автоматизованих системах

спостереження, розвідки та аналітики зображень, зокрема в умовах воєнного або надзвичайного середовища.

Особливої уваги заслуговує остання версія архітектури – YOLOv8, яка втілює низку удосконалень, спрямованих на ще більшу продуктивність і точність. У порівнянні з попередніми реалізаціями, YOLOv8 використовує покращену структуру Backbone на базі CSPNet, модернізовані шари нормалізації, вдосконалені функції втрат та адаптивніші механізми передбачення. У YOLOv8 реалізовано підтримку як цілочисельних, так і плаваючих ваг, що відкриває можливості для використання на різноманітних обчислювальних платформах – від GPU до мобільних процесорів і edge-пристроїв.

Крім того, YOLOv8 підтримує гнучке масштабування моделі завдяки різним конфігураціям (наприклад, YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l та YOLOv8x), що дозволяє обрати оптимальний баланс між швидкістю та точністю відповідно до конкретного сценарію застосування. Наприклад, у проєктах виявлення фортифікаційних споруд з БПЛА, де критичною є обробка великих обсягів аерофотоінформації в режимі близькому до реального часу, YOLOv8s або YOLOv8m забезпечують оптимальний компроміс між ефективністю та якістю результату. У той же час, старші версії моделі можуть бути використані для більш точного аналізу у фазі постобробки.

YOLOv8 також демонструє підвищену здатність до генералізації, що особливо важливо при роботі з нестандартними або незбалансованими наборами даних, як це часто буває при обробці зображень військової тематики. Її використання дозволяє автоматизувати аналіз зображень, прискорити прийняття рішень у польових умовах і значно зменшити навантаження на операторів систем моніторингу. Таким чином, YOLOv8 є не лише еволюційним кроком у розвитку архітектур глибокого навчання для виявлення об'єктів, а й потужним інструментом у вирішенні актуальних безпекових та оборонних задач.

## 2.4. Основні кроки методу

Процес донавчання моделі YOLOv8 для задачі виявлення фортифікаційних споруд передбачає низку послідовних етапів, спрямованих на адаптацію базової моделі до специфіки нових даних. Загальна структура цього процесу представлена у вигляді трьох логічних блоків: вхідні дані, обробка даних та вихідні дані (рисунок 2.4).



Рисунок 2.4 – Схема методу донавчання моделі YOLOv8 для виявлення фортифікаційних споруд

На етапі підготовки вхідних даних формується навчальний датасет, який включає зображення фортифікаційних споруд, що мають відповідну розмітку (bounding boxes) для коректного навчання моделі. Даний датасет формується із використанням супутникових, аерофотознімків або інших візуальних матеріалів, де присутні об'єкти, які можна віднести до класу фортифікаційних. Як початкова основа для донавчання використовується попередньо натренована модель YOLOv8 – сучасна архітектура згорткових нейронних мереж, призначена для задач виявлення об'єктів у реальному часі.

Після етапу формування вхідних даних здійснюється їх обробка, яка складається з двох основних кроків. Перший крок передбачає попередню обробку зображень – зокрема, зміну розміру, нормалізацію піксельних значень, а також застосування методів аугментації (таких як обертання, масштабування, віддзеркалення тощо). Це дозволяє покращити стійкість моделі до змін умов зйомки та підвищити її здатність до узагальнення.

Другий крок полягає у безпосередньому донавчанні (fine tuning) моделі YOLOv8 на новому датасеті. У ході цього процесу модель адаптує свої параметри під нову вибірку, що дозволяє суттєво підвищити точність виявлення специфічних об'єктів – у даному випадку фортифікаційних споруд. Доновчання виконується з використанням алгоритму зворотного поширення помилки, і передбачає часткове або повне оновлення ваг нейромережі залежно від попередньої ступені її спеціалізації.

Завершальним етапом є отримання донавченої моделі YOLOv8, яка здатна із високою точністю розпізнавати фортифікаційні об'єкти на нових зображеннях. Така модель може бути інтегрована у системи автоматизованого аналізу зображень для використання в оборонній сфері, інфраструктурному моніторингу або при проведенні інженерної розвідки.

У межах реалізації задачі автоматичного виявлення фортифікаційних споруд наступним логічним етапом стало безпосереднє використання донавченої моделі YOLOv8 для обробки зображень. Застосування цієї моделі передбачає послідовне виконання ряду операцій, що представлені на схемі (рисунок 2.5).



Рисунок 2.5 – Схема використання донавченої моделі для детекції фортифікаційних споруд.

Вхідними даними є натренована модель, здатна розпізнавати відповідні класи об'єктів, а також зображення, на яких потенційно можуть бути присутні фортифікаційні споруди. Процес обробки зображень відбувається у кілька послідовних етапів.

На першому кроці здійснюється завантаження вхідних зображень та їх попередня обробка, що включає масштабування, нормалізацію та приведення до

формату, сумісного з архітектурою моделі YOLOv8. Наступним кроком є виявлення об'єктів, у ході якого модель виконує інференс і локалізує об'єкти, що належать до цільового класу – фортифікаційних споруд. На завершальному етапі формуються обмежувальні рамки (bounding boxes) навколо виявлених об'єктів, які згодом накладаються на зображення для візуалізації результатів детекції.

Результатом цього процесу є зображення з чітко окресленими прямокутниками, що позначають знайдені елементи інженерної інфраструктури. Таким чином, схема ілюструє типову послідовність дій при застосуванні донавченої моделі глибокого навчання для вирішення прикладної задачі об'єктної детекції в контексті оборонного аналізу.

Реалізація методу донавчання моделі YOLOv8 для виявлення фортифікаційних споруд демонструє ефективність сучасних підходів глибокого навчання у вирішенні прикладних задач об'єктної детекції. Завдяки високій точності та швидкості обробки зображень, ця модель забезпечує надійне виявлення цільових об'єктів навіть у складних умовах. Послідовність етапів – від підготовки та анотації даних до донавчання та застосування моделі – дозволяє адаптувати універсальну архітектуру YOLOv8 до специфічних потреб завдання. Таким чином, використання донавченої моделі YOLOv8 є доцільним та ефективним підходом для автоматизованого виявлення фортифікаційних споруд на зображеннях, отриманих з різних джерел.

## **2.5. Метрики оцінювання**

З метою оцінки якості роботи нейронної мережі для виявлення траншей на зображеннях, отриманих із безпілотних літальних апаратів, було проведено аналіз основних метрик якості класифікації та локалізації об'єктів. Особливу увагу приділено F1-мірі, точності (precision), повноті (recall) та середній точності (mAP). Для зручності візуального аналізу представлено приклади відповідних графіків: F1-Confidence Curve, Recall-Confidence Curve та Precision-Recall Curve.

## Графік залежності F1-міри від порогу впевненості (F1-Confidence Curve)

Даний графік (рис. 2.6) відображає зміну значення F1-міри в залежності від встановленого порогу впевненості, за якого модель вважає об'єкт виявленим. F1-міра є одним із ключових інтегральних показників якості класифікаційних та детекційних моделей, що дозволяє об'єктивно оцінити їхню ефективність в умовах, коли дані містять незбалансовану кількість представників різних класів. Цей показник являє собою гармонійне середнє між двома важливими метриками – точністю (precision) та повнотою (recall), що відображають відповідно частку правильно ідентифікованих позитивних результатів серед усіх передбачених як позитивні, та частку правильно виявлених позитивних результатів серед усіх дійсно позитивних прикладів у вибірці.

Завдяки об'єднанню цих двох характеристик, F1-міра дозволяє враховувати як хибнопозитивні, так і хибнонегативні передбачення, тим самим забезпечуючи більш збалансовану оцінку продуктивності моделі, особливо в тих випадках, коли одна з метрик (точність або повнота) має суттєве домінування над іншою.

Такий підхід є особливо актуальним у прикладних завданнях, де важлива як здатність моделі не пропускати об'єкти (висока повнота), так і уникати хибного виявлення неіснуючих об'єктів (висока точність) [32]. В контексті виявлення фортифікаційних споруд це має критичне значення, адже пропущений об'єкт може мати серйозні наслідки для прийняття оперативних рішень, тоді як зайві помилкові спрацювання можуть призвести до марного витрачання ресурсів. F1-міра, як збалансований показник, дозволяє обрати оптимальний поріг впевненості моделі, що забезпечує найкраще співвідношення між точністю і повнотою.

Візуалізація цієї залежності у вигляді графіка F1-міри допомагає краще зрозуміти поведінку моделі при зміні її чутливості до виявлення об'єктів, а також сприяє обґрунтованому вибору параметрів для реального застосування. Таким чином, аналіз F1-Confidence Curve є важливим етапом оцінки надійності моделі в умовах складного середовища.

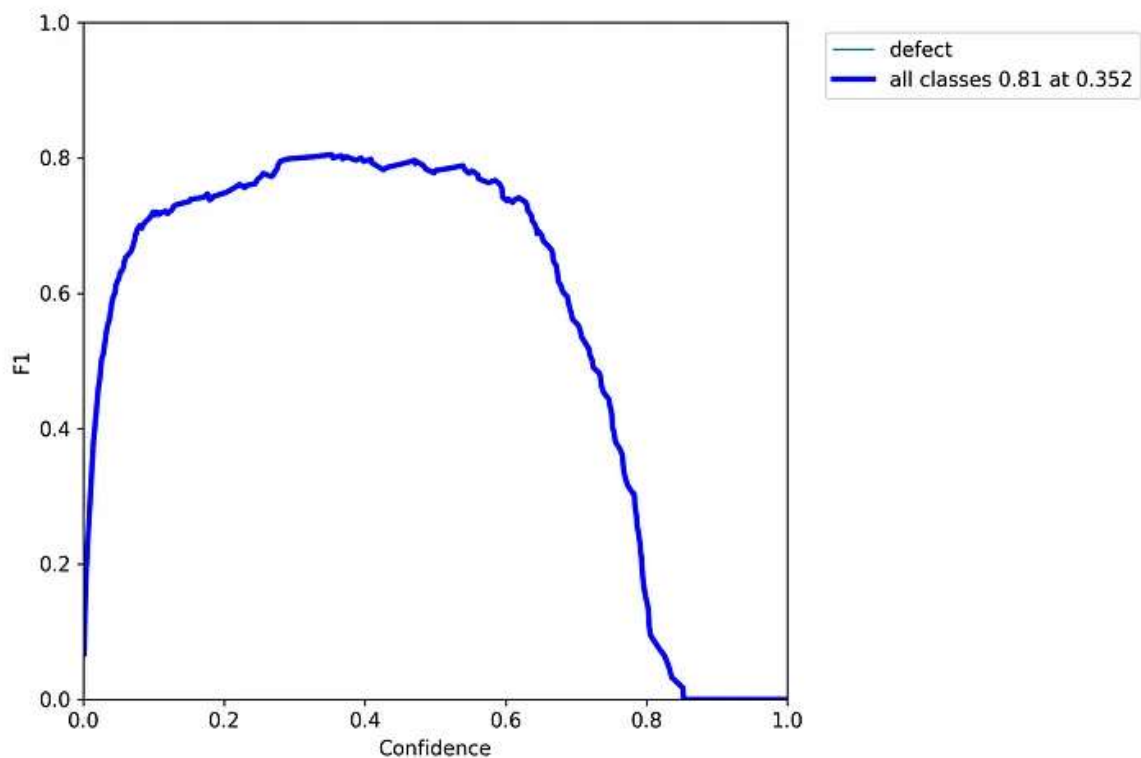


Рисунок 2.6 – Приклад графіку F1-Confidence Curve [32]

### Графік залежності recall від порогу впевненості (Recall-Confidence Curve)

Графік залежності recall від порогу впевненості (Recall-Confidence Curve) є важливим інструментом для аналізу ефективності класифікаційних моделей. Цей графік демонструє, як змінюється чутливість моделі, або recall, в залежності від порогу впевненості, який визначає, наскільки переконаним має бути алгоритм, щоб класифікувати об'єкт як позитивний.

У контексті класифікації, recall описує здатність моделі правильно знаходити позитивні приклади. Зазвичай, чим нижчий поріг впевненості, тим більше прикладів модель буде класифікувати як позитивні, що веде до збільшення recall. Однак, при занадто низькому порозі, модель може помилково класифікувати багато негативних прикладів як позитивні, що, у свою чергу, може призвести до зниження precision.

На графіку Recall-Confidence Curve по осі X відкладаються значення порогу впевненості, а по осі Y – значення recall. Як правило, при високому порозі модель класифікує лише найбільш очевидні позитивні приклади, і тому recall є

низьким. При зниженні порогу модель стає більш ліберальною в класифікації, тому recall починає зростати. Проте, якщо поріг буде занадто низьким, збільшення recall може супроводжуватися високим рівнем помилкових позитивних класифікацій [33].

Такий графік допомагає зрозуміти компроміс між recall та іншими показниками моделі, такими як precision. Це важливо для вибору оптимального порогу, який відповідає специфічним вимогам завдання, наприклад, коли важливо мінімізувати кількість пропущених позитивних прикладів, навіть якщо це призведе до збільшення помилкових спрацьовувань.

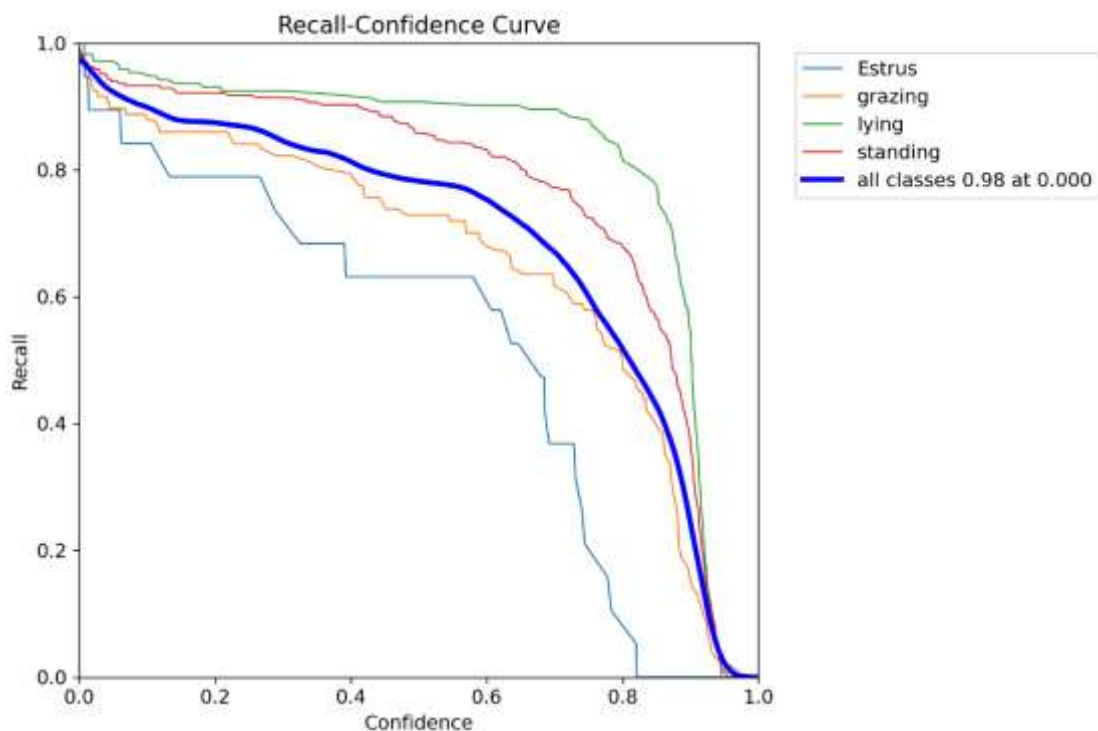


Рисунок 2.7 – Приклад графіку Recall-Confidence [33]

Отже, обране значення порогу має бути компромісом між охопленням усіх об'єктів (високий recall) та мінімізацією хибнопозитивних результатів (високий precision).

### **Графік залежності точності від повноти (Precision-Recall Curve)**

Графік залежності точності від повноти (Precision-Recall Curve) є важливим інструментом для оцінки ефективності класифікаційних моделей,

особливо в умовах дисбалансу класів, коли позитивні приклади є рідкісними або важкими для виявлення. Цей графік демонструє компроміс між двома ключовими метриками – точністю (precision) та повнотою (recall) – при зміні порогу впевненості, що використовується для класифікації позитивних прикладів.

Точність визначається як відношення кількості правильно класифікованих позитивних прикладів до загальної кількості прикладів, які модель класифікувала як позитивні. Повнота, в свою чергу, вказує на здатність моделі знаходити всі позитивні приклади серед усіх дійсно позитивних об'єктів. Графік Precision-Recall Curve дозволяє візуалізувати, як ці дві метрики змінюються в залежності від порогу, що використовується для класифікації [33].

У разі задачі виявлення фортифікаційних споруд, графік Precision-Recall Curve дозволяє оцінити ефективність моделі в умовах, коли точність та повнота мають важливе значення для прийняття рішень. Наприклад, складний рельєф або схожість деяких природних елементів з траншеями може призвести до помилок класифікації, що знижує точність моделі.

Графік Precision-Recall Curve дозволяє наочно побачити, як варіація порогу впевненості змінює взаємозв'язок між точністю та повнотою. При високому порозі модель буде класифікувати лише найбільш очевидні позитивні приклади, що може призвести до високої точності, але знизить recall, оскільки деякі траншеї будуть пропущені. Зниження порогу може збільшити recall, оскільки модель буде більше обережно класифікувати приклади як позитивні, однак при цьому точність може знизитися через збільшення кількості хибних позитивних класифікацій.

Графік Precision-Recall (рис. 2.8) дозволяє оцінити загальну продуктивність моделі при зміні порогу впевненості. Крива має типовий спадний характер, що відображає компроміс між precision і recall: із зростанням повноти точність, як правило, знижується.

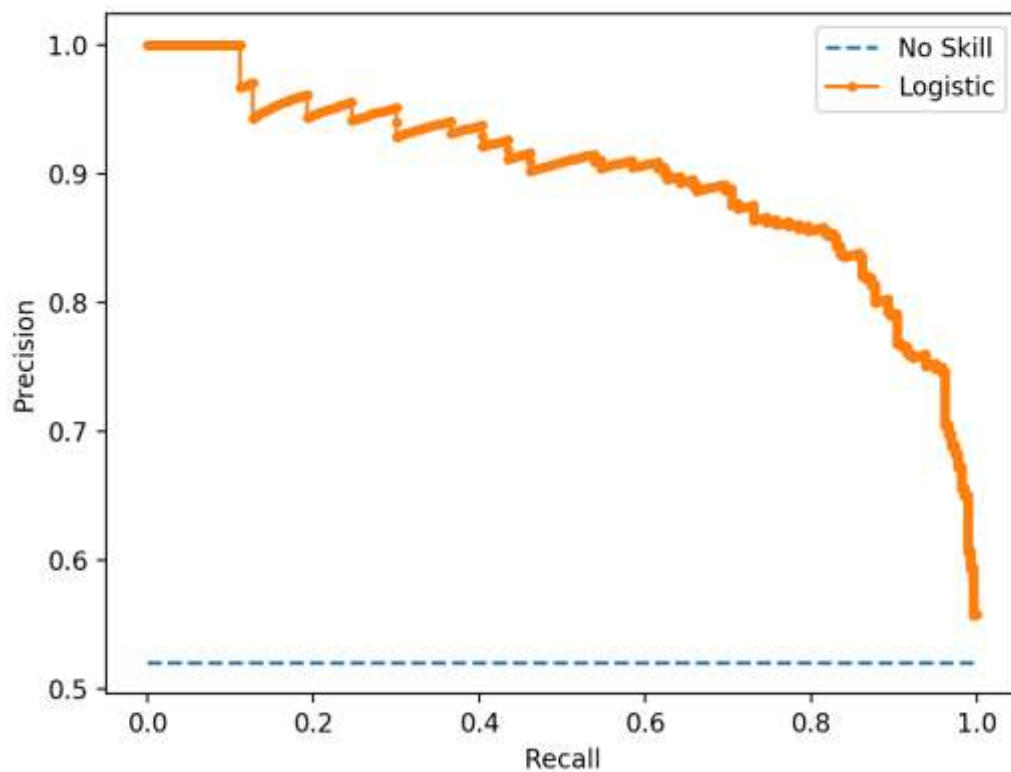


Рисунок 2.8 – Приклад графіку Precision-Recall [33]

### Оцінка динаміки навчання моделі

На рисунку 2.9 зображено приклади графіків основних функцій втрат (loss) та метрик точності (precision, recall, mAP). Аналіз таких графіків є важливою частиною оцінки ефективності моделі, оскільки дозволяє простежити динаміку навчання, виявити потенційні проблеми (перенавчання, високі втрати тощо) і зробити висновки щодо загальної придатності моделі до практичного застосування.

Для комплексного аналізу були враховані як тренувальні, так і валідаційні характеристики, зокрема втрати за компонентами локалізації та класифікації, а також метрики точності виявлення [34].

Особливу увагу було приділено стабільності показників на валідаційному наборі, оскільки саме вони свідчать про узагальнюючу здатність моделі.

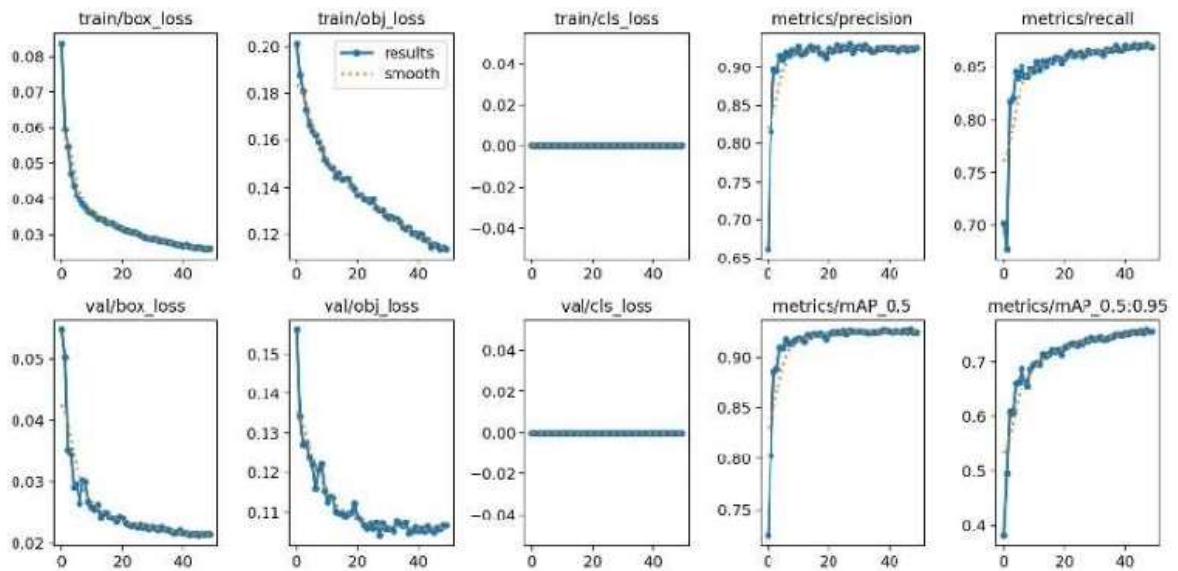


Рисунок 2.9 – Приклади графіків функцій втрат [34]

Для оцінки ефективності навчання моделі глибокого навчання, зокрема в задачах виявлення об'єктів, широко використовуються графіки зміни ключових метрик, таких як Precision (точність), Recall (повнота) та mAP (mean Average Precision – середнє значення точності). Ці графіки відображають, як змінюється якість роботи моделі впродовж епох навчання, і слугують інструментом для контролю збіжності та ефективності процесу оптимізації.

Точність (Precision) вимірює частку правильно виявлених об'єктів серед усіх, які модель класифікувала як позитивні. Високе значення цієї метрики свідчить про низький рівень хибнопозитивних спрацювань.

Повнота (Recall) визначає частку правильно виявлених об'єктів серед усіх фактично наявних на зображеннях. Вона показує здатність моделі знаходити всі релевантні об'єкти, мінімізуючи хибновідмови.

mAP (mean Average Precision) є інтегральною метрикою, яка поєднує точність і повноту для різних порогів впевненості моделі, забезпечуючи комплексну оцінку її загальної ефективності.

У процесі тренування очікується, що всі три показники зростатимуть, свідчаючи про покращення здатності моделі виявляти об'єкти точно і повно. Графіки дозволяють своєчасно виявити перенавчання, недонавчання або стагнацію навчання. Наприклад, стабілізація mAP на високому рівні зазвичай

свідчить про досягнення оптимальної якості моделі. Таким чином, аналіз цих графіків є важливою складовою обґрунтування працездатності та доцільності застосування запропонованого методу.

### Матриця кореляцій

Матриці кореляцій застосовуються як інструмент аналітичної оцінки у процесі розробки та навчання моделей комп'ютерного зору, зокрема при вирішенні задач детектування об'єктів (рис. 2.10).

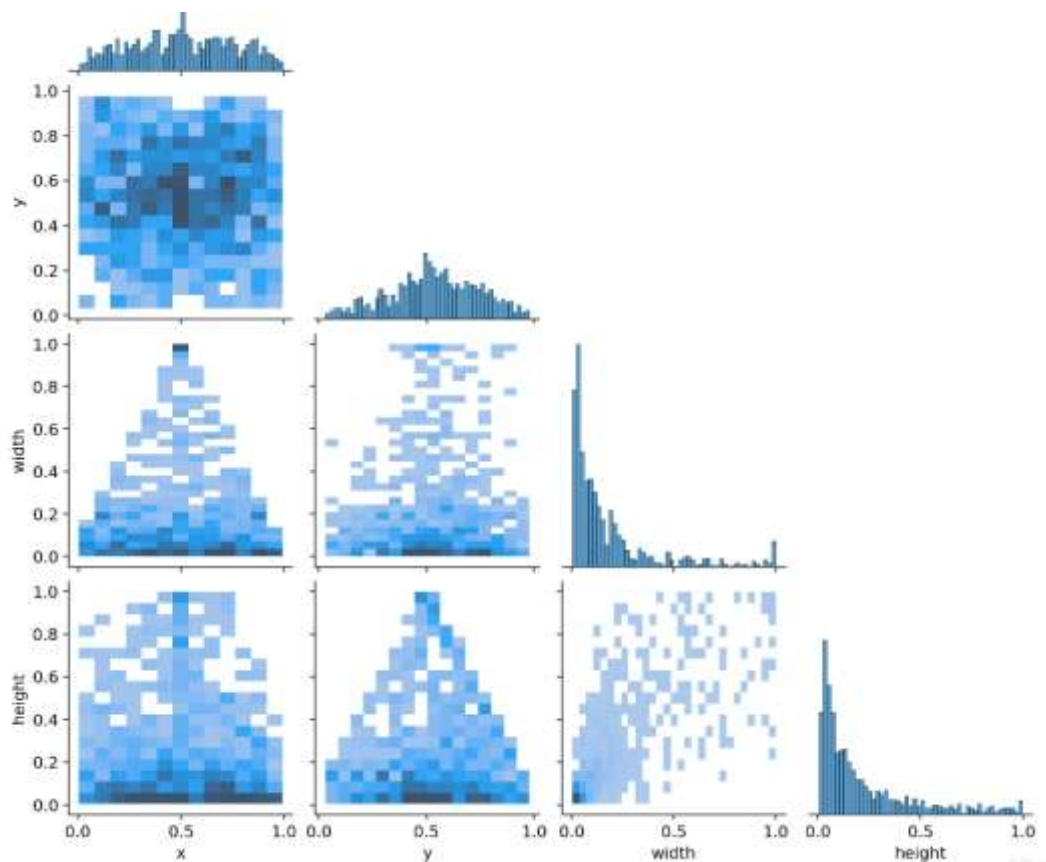


Рисунок 2.10 – Приклад матриці кореляцій [34]

Вони дозволяють встановити ступінь взаємозв'язку між різними параметрами, метриками або компонентами функцій втрат, що формуються в ході тренування моделі. Такий підхід сприяє глибшому розумінню внутрішньої динаміки навчання та виявленню потенційних проблем на ранніх етапах.

Зокрема, матриці кореляцій можуть бути корисними для аналізу взаємозалежності між метриками точності (precision, recall, mAP) і втратами (наприклад, втрати локалізації, класифікації, об'єктності). Висока позитивна або

негативна кореляція між окремими показниками може свідчити про сильний вплив одних компонентів на інші, що може призводити до нестабільного навчання або переорієнтації моделі на окремі аспекти задачі.

Крім того, кореляційний аналіз дає змогу виявити надмірні або дубльовані ознаки вхідних даних, а також визначити можливі конфлікти між метриками, наприклад, між точністю та повнотою. Це особливо актуально в умовах високої складності датасетів, наявності шумів або нерівномірного розподілу класів.

Узагальнюючи, використання матриць кореляцій дозволяє підвищити якість інтерпретації результатів навчання, оптимізувати процес калібрування гіперпараметрів і приймати обґрунтовані рішення щодо модифікації архітектури або структури навчальних даних.

## 2.7 Висновки до розділу 2

У ході дослідження було запропоновано метод виявлення фортифікаційних споруд на зображеннях, отриманих із безпілотних літальних апаратів, із використанням сучасних засобів глибокого навчання. Основу розробки склала модель YOLOv8, яка поєднує високу швидкодію, компактність архітектури та конкурентну точність. Особливість методу полягає в адаптації попередньо натренованої моделі до специфіки фортифікаційних об'єктів завдяки донавчанню (fine-tuning) на спеціалізованому датасеті.

Для реалізації цієї мети було використано датасет Kalumbia\_map, який містить понад 500 зображень із розмітками типових оборонних об'єктів – траншей, окопів, бліндажів тощо. Дані було підготовлено з використанням платформи Roboflow, що забезпечила ефективну анотацію у вигляді bounding box-ів. Розподіл вибірки на тренувальну та валідаційну частини дозволив реалізувати повноцінний цикл навчання та оцінки моделі, зберігаючи узагальнюючу здатність і мінімізуючи ризики перенавчання.

Отже, було представлено цілісну концепцію методу донавчання моделі YOLOv8 для задачі автоматизованого виявлення фортифікаційних споруд, що

включає теоретичне обґрунтування, структуровану методологію реалізації та практичні аспекти використання моделі. Детально розглянуто всі етапи – від формування навчального датасету, який містить якісно анотовані зображення, до процесу обробки вхідних даних і проведення безпосереднього донавчання. Продемонстровано важливість попередньої обробки та аугментації зображень для забезпечення стійкості моделі до варіативних умов зйомки, а також описано специфіку адаптації ваг згорткової нейронної мережі до нових даних.

Проаналізовано алгоритм використання донавченої моделі для обробки візуальних матеріалів, отриманих із різних джерел, таких як безпілотні літальні апарати. Запропонована схема охоплює всі технічні кроки – від завантаження зображень і їх перетворення до візуалізації результатів виявлення об'єктів на основі bounding boxes. У результаті отримується інструмент, що може бути інтегрований у системи спостереження, оборонного аналізу або технічного моніторингу.

В межах розділу також обґрунтовано вибір архітектури YOLOv8 як базової платформи, що забезпечує баланс між точністю, швидкістю та адаптивністю. Значну увагу приділено системі оцінювання якості моделі, зокрема застосуванню ключових метрик (precision, recall, F1-score, mAP), візуалізаційних графіків (F1-Confidence, Recall-Confidence, Precision-Recall), а також аналізу динаміки функцій втрат під час тренування. Додатково розглянуто використання кореляційних матриць як інструменту діагностики, що дозволяє виявляти взаємозв'язки між компонентами моделі та уточнювати стратегії її оптимізації.

Таким чином, викладені в розділі матеріали формують цілісне теоретико-практичне підґрунтя для створення та розгортання високоефективної системи виявлення фортифікаційних споруд на основі глибокого навчання.

## **Розділ 3 Експериментальне дослідження запропонованого методу**

### **3.1 Опис застосування для експериментального дослідження**

#### **3.1.1. Бібліотеки та фреймворки для створення застосування**

Для створення високоточних, адаптивних і масштабованих рішень, здатних до обробки великих обсягів візуальної інформації в умовах, наближених до реального часу, потрібно запропонувати відповідні високоефективні програмні рішення та технології.

Основним фреймворком для побудови та навчання глибоких нейронних мереж був вибраний PyTorch [35]. Його динамічна побудова обчислювального графа надає значну гнучкість у процесі моделювання архітектури та проведення експериментів, що є надзвичайно важливим на етапах тестування гіпотез щодо структури моделі. Додатковими перевагами є висока продуктивність на GPU, що значно зменшує час навчання моделей, а також потужна екосистема інструментів для дебагу, моніторингу та візуалізації. PyTorch має відкриту архітектуру, яка дозволяє безперешкодно комбінувати його з іншими інструментами (наприклад, OpenCV [36] або бібліотеками анотації зображень). Серед потенційних недоліків слід зазначити вищий поріг входження для початківців у порівнянні з TensorFlow [37] у режимі Keras, однак для даної роботи цей аспект був нівельований наявністю попереднього досвіду роботи з PyTorch. Таким чином, саме ця технологія є найбільш релевантною до завдання, що вимагає динамічного контролю та високої точності під час локалізації об'єктів.

Базовою моделлю для задачі детекції об'єктів була обрана YOLOv8 (You Only Look Once, версія 8) [31]. Однією з ключових причин її вибору є здатність виконувати розпізнавання в реальному часі з високим рівнем точності. YOLOv8 поєднує переваги легких архітектур з потужністю сучасних глибоких моделей, зберігаючи при цьому оптимальний баланс між швидкістю та якістю детекції. У контексті роботи з БПЛА важливо мати модель, яка здатна оперативно обробляти потік зображень у польових умовах – навіть за обмежених обчислювальних

ресурсів. Модель підтримує багатокласову класифікацію, що дозволяє виявляти різні типи фортифікаційних споруд на основі їхнього візуального профілю. З-поміж недоліків YOLOv8 можна виокремити складність налаштування під специфічні датасети та потребу в ретельному маркуванні зображень. Водночас наявність відкритої документації, активної спільноти та прикладів впровадження компенсує ці виклики. Вибір цієї моделі є доцільним, оскільки саме YOLOv8 забезпечує необхідну точність і продуктивність для військових та аналітичних застосувань, де час прийняття рішення має критичне значення.

Для попередньої обробки зображень, зокрема фільтрації шумів, нормалізації контрасту, геометричних трансформацій і побудови контурів було вибрано OpenCV (Open Source Computer Vision Library) [36]. Висока продуктивність і сумісність із іншими бібліотеками Python робить її практично незамінною при підготовці зображень до подальшої обробки моделлю. Бібліотека також дозволяє проводити візуалізацію результатів через побудову обмежувальних прямокутників (bounding boxes), що сприяє оцінці якості моделі. До недоліків можна віднести обмежену інтеграцію з глибокими нейронними мережами (вона не є фреймворком глибокого навчання), однак у поєднанні з PyTorch та YOLOv8 цей недолік нівелюється. Використання OpenCV в контексті даної роботи є обґрунтованим з огляду на її роль як інструмента для підготовки даних і візуального аналізу ефективності моделей.

Узагальнюючи, можна констатувати, що кожен із обраних інструментів виконує ключову функцію в реалізації дослідження: PyTorch – забезпечує гнучкість і точність нейронних мереж, YOLOv8 – забезпечує ефективність детекції в реальному часі, OpenCV – забезпечує якісну обробку зображень. Комплексне застосування цих технологій дозволило побудувати технологічно стійкий, продуктивний та масштабований застосунок, здатний до реального використання в умовах, що змінюються, та адаптації до нових задач у сфері оборонної безпеки.

### 3.1.2. Діаграма класів застосування

Для експериментального дослідження запропонованого методу виявлення фортифікаційних споруд на зображеннях, отриманих за допомогою БПЛА, було розроблено відповідну діаграму класів (рисунок 3.1) для графічного відображення взаємозв'язків класів у застосуванні та кращого розуміння її архітектури.

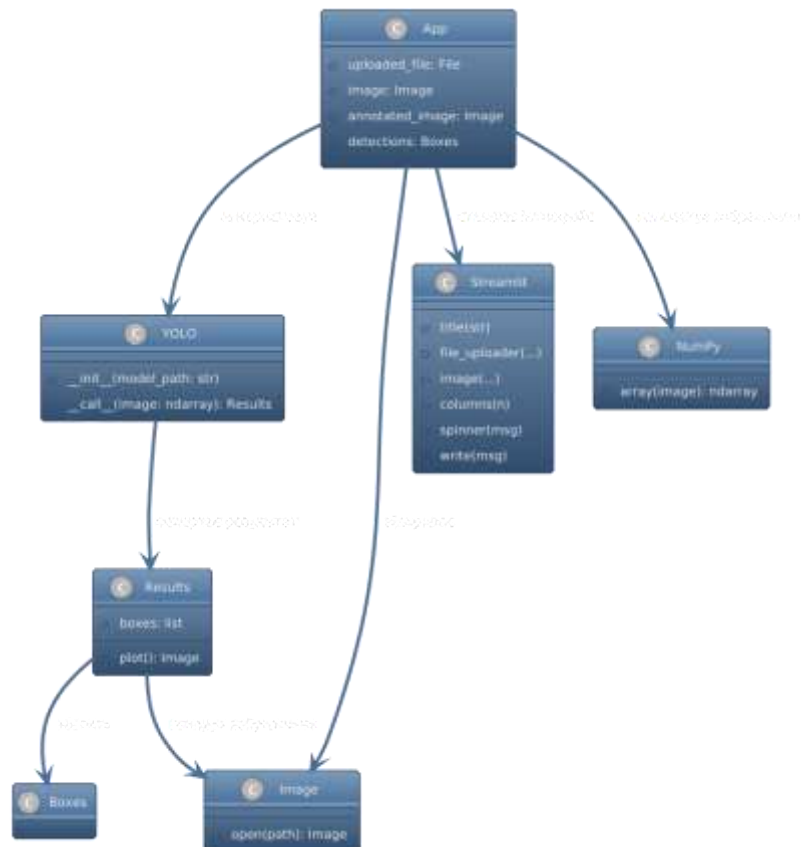


Рисунок 3.1 – Графічне відображення діаграми класів застосування

Діаграма класів моделює основні компоненти програмного застосунку, який реалізує запропонований метод. Основна функціональність реалізована в середовищі Python із використанням бібліотек Streamlit [38], Ultralytics YOLO [39], NumPy [40] та Pillow [41].

Клас App не реалізований явно в кодї, але його можна трактувати як умовний контейнер, що координує взаємодію між іншими класами та модулями. Він відповідає за створення веб-інтерфейсу через Streamlit, завантаження

зображення користувачем, виклик функції обробки `process_image` та відображення результатів.

Клас `Yolo` – це клас бібліотеки `Ultralytics`, який інкапсулює завантажену модель глибокого навчання у форматі `.pt` (`PyTorch`). Його основні методи – «`__initt__(model_path)`» (завантажує модель за вказаним шляхом), «`_call__(image)`» (обробляє зображення і повертає результат детекції). У діаграмі класів він пов’язаний із класом `App` (як користувач функціональності моделі) та класом `Results` (результат обробки).

Клас `Results` – це клас, який містить результат виконання `YOLO`-моделі, тобто це об’єкт, що містить список виявлених об’єктів (`boxes`) та візуалізує результат через метод `plot`, який повертає анотоване зображення. Цей клас є критичним для взаємодії між ядром обчислень і представленням даних у візуальній формі.

Клас `Image` представляє зображення, що завантажується користувачем. Створюється за допомогою «`Image,open()`» з бібліотеки `Pillow`. Перед передачею моделі конвертується в масив `NumPy` для сумісності з `PyTorch`.

Клас `NumPy` здійснює низькорівневе перетворення зображення з формату `PIL.image` до `ndarray` за допомогою `np.array(image)`. Це потрібно для того, що модель могла працювати із зображенням як із тензором.

Клас `Streamlit` – це бібліотека для створення інтерфейсу користувача.

Таким чином спроектовану програмну структуру застосування та описано функціональне призначення її складових.

### **3.1.3 Особливості реалізації програмних складових системи**

Програмна реалізація системи автоматичного виявлення фортифікаційних споруд на зображеннях з безпілотних літальних апаратів (БПЛА) побудована на принципах модульного програмування [42], що передбачає розділення всієї системи на окремі, логічно завершені компоненти. Кожен з компонентів виконує чітко визначену функцію та взаємодіє з іншими через стандартизовані інтерфейси.

Такий підхід дозволяє забезпечити не лише надійність функціонування програми, а й її гнучкість, масштабованість та зручність супроводу.

Архітектура системи включає п'ять основних логічних компонентів:

- головний застосунок (App) координує роботу всіх інших модулів;
- інтерфейс користувача (Streamlit) відповідає за взаємодію з користувачем;
- модуль нейронної мережі (YOLO) здійснює аналіз зображень;
- модуль обробки результатів (Results і Boxes) інкапсулює результати обробки;
- модулі зображень та масивів (Image, NumPy) забезпечують роботу з графічними даними.

### **Головний застосунок (App)**

Клас App виконує роль центрального диспетчера системи. У ньому зосереджено логіку взаємодії користувача з інтерфейсом, завантаження вхідних даних, передача зображень до нейронної мережі, а також обробка та візуалізація результатів. Серед основних змінних, що входять до його складу:

`uploaded_file` – об'єкт, що містить завантажене користувачем зображення у вигляді файлу;

`image` – конвертоване зображення у форматі, придатному для аналізу;

`annotated_image` – результат обробки нейронною мережею із нанесеними візуальними позначками;

`detections` – набір об'єктів типу `Boxes`, які містять координати знайдених об'єктів та супутню інформацію.

Фактично, саме в межах цього класу здійснюється повна оркестрація усіх етапів роботи програми: від отримання вхідних даних – до виводу кінцевого зображення з позначеними фортифікаційними спорудами.

### **Компонент графічного інтерфейсу (Streamlit)**

Для реалізації зручного та зрозумілого інтерфейсу користувача було використано фреймворк Streamlit, який дозволяє швидко створювати інтерактивні веб-застосунки засобами Python. Цей компонент забезпечує:

- формування заголовків (`title()`);
- завантаження файлів користувачем (`file_uploader()`);
- виведення зображень (`image()`);
- створення динамічних колонок (`columns()`);
- повідомлення про хід обробки (`spinner()`);
- текстовий супровід результатів (`write()`).

Інтерфейс побудований таким чином, щоб зберігати максимальну простоту: користувач лише завантажує зображення, після чого система автоматично виконує аналіз і повертає результат у вигляді графічного зображення з детекціями.

### **Компонент глибинного навчання (YOLO)**

Клас YOLO реалізує інтерфейс до попередньо навченої моделі глибинного навчання YOLOv8. При ініціалізації екземпляра класу здійснюється завантаження збережених ваг моделі (файл `.pt` або `.onnx`) для подальшого використання.

Основною функцією цього модуля є метод `call()`, який приймає зображення у вигляді масиву NumPy (тензора) і повертає результат обробки у вигляді об'єкта `Results`. На виході отримуються координати знайдених об'єктів (`bounding boxes`), їх класи та рівень впевненості, з яким модель виявила відповідний об'єкт.

Серед ключових особливостей:

- підтримка високої швидкості обробки зображень;
- здатність працювати із зображеннями високої роздільної здатності;
- точна локалізація об'єктів за допомогою `anchor-based` регресії.

### **Модуль результатів обробки (Results та Boxes)**

Після обробки зображення нейромережею, дані про виявлені об'єкти інкапсулюються у спеціальний об'єкт `Results`. Він містить:

- список об'єктів `boxes`, кожен з яких представляє координати виявленого об'єкта;
- метод `plot()`, який дозволяє нанести виявлені межі об'єктів на зображення та повернути його у зручному для відображення форматі.

Об'єкт `Voxes`, у свою чергу, містить координати, клас об'єкта (наприклад, «бліндаж», «окоп» тощо) та значення впевненості. Цей модуль відіграє важливу роль у формуванні вихідного зображення та подальшій фільтрації об'єктів за значенням порогу впевненості.

### **Модулі обробки зображень (Image) та числових даних (NumPy)**

Зображення, що обробляються системою, відкриваються за допомогою класу `Image`, який забезпечує читання зображення із заданого шляху та приведення його до необхідного формату. У випадку попередньої обробки або модифікації зображення (наприклад, зміна розміру, обертання) використовується бібліотека `OpenCV`.

Числові обчислення та передача даних до нейронної мережі виконуються з використанням масивів типу `ndarray`, які формуються за допомогою бібліотеки `NumPy`. Саме в такому форматі дані подаються на вхід моделі `YOLO`.

Однією з ключових рис реалізації стала орієнтація на безперервну обробку даних у режимі реального часу, що вимагає стабільної взаємодії між обчислювальними модулями та інтерфейсною частиною. Такий підхід дозволив досягти високої оперативності у виконанні завдань, пов'язаних із виявленням об'єктів військового значення на основі фотозображень. У процесі створення застосунку значну увагу було приділено внутрішній архітектурі, яка забезпечує гнучке масштабування та можливість подальшого удосконалення без радикальних змін у базовій структурі коду. Компоненти системи логічно відокремлені відповідно до функціонального призначення, що сприяє зрозумілості коду та спрощує супровід у майбутньому.

Особливістю реалізації стала інтеграція попередньо натренованої нейронної мережі `YOLOv8` (рисунок 3.2), яка була адаптована до поставленої задачі через тонке налаштування параметрів та підбір власного датасету з урахуванням специфіки знімків. Реалізація механізму обробки передбачала етапи декодування зображень, нормалізації вхідних даних, передачі масивів до нейромережі, інтерпретації її вихідних параметрів, після чого здійснюється формування фінального результату. Такий ланцюг дій реалізований у вигляді

компактної логіки в основному модулі програми, з урахуванням усіх виняткових ситуацій, пов'язаних з можливою помилкою обробки або недоступністю ресурсу.

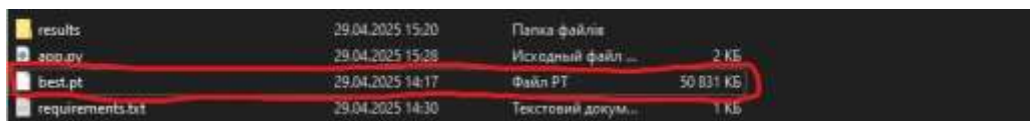


Рисунок 3.2 – попередньо натренована модель YOLO

Застосунок вирізняється також увагою до питань інтерфейсної взаємодії, яка була реалізована засобами швидкого веб-рендерингу. Хоча зовнішня частина системи не містить складної анімації чи багатоступеневої логіки, її структура чітко відповідає принципам мінімалізму та інтуїтивності. Розроблений вебзастосунок для виявлення фортифікаційних споруд на зображеннях з БПЛА реалізовано на основі бібліотеки Streamlit та попередньо натренованої моделі YOLO.

Для роботи системи передбачено попереднє налаштування середовища, яке виконується один раз. Зокрема, встановлюється дистрибутив Anaconda та завантажується архів із файлами моделі. Ініціалізація здійснюється через спеціалізовану консоль Anaconda Prompt, інтерфейс якої подано на рисунку 3.3.

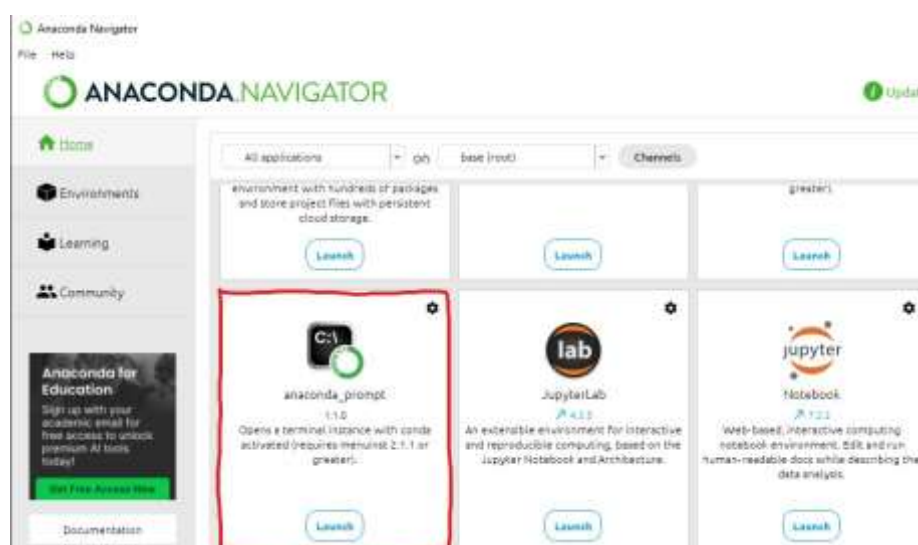


Рисунок 3.3 – Інтерфейс дистрибутиву Anaconda

У середовищі створюється окреме віртуальне оточення, встановлюється необхідна версія Python та потрібні бібліотеки (зокрема, streamlit, ultralytics, opencv-python-headless, Pillow, numpy). Після цього система готова до запуску.

Увімкнений застосунок відкривається у веб браузері (рисунок 3.4). Інтерфейс є інтуїтивно зрозумілим: користувачу достатньо натиснути кнопку «Browse files» і завантажити зображення, отримане з безпілотного літального апарата.

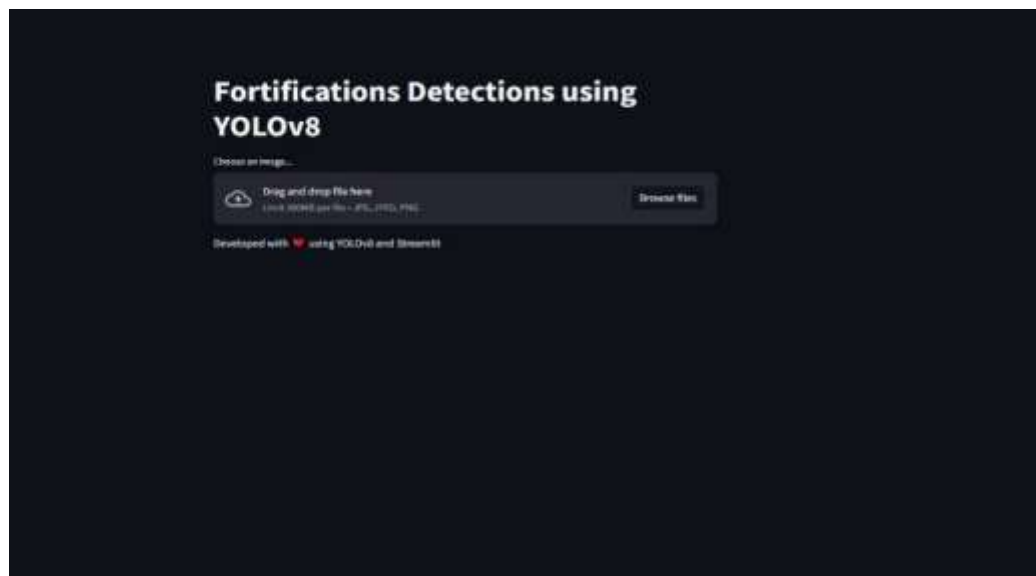


Рисунок 3.4 – Інтерфейс вебзастосунку

Після завантаження зображення система автоматично проводить аналіз із використанням алгоритмів комп'ютерного зору. Як результат, користувач отримує візуалізоване зображення з виділеними ділянками, які ідентифіковано як фортифікаційні споруди (рисунок 3.5).

На зображенні автоматично відображаються обмежувальні прямокутники та відповідні оцінки ймовірності, що дозволяє оперативно оцінити результат. Такий формат подання є зручним для швидкої інтерпретації, навіть для користувачів без спеціальних технічних знань.

Таким чином, створений вебзастосунок забезпечує простий та ефективний інтерфейс для практичного застосування методу в реальних умовах.



Рисунок 3.5 – Результат роботи виявлення фортифікаційних споруд

Таким чином, розроблений метод демонструє приклад успішної інтеграції методів комп'ютерного зору, глибинного навчання та сучасних веб-технологій. Чітке розділення компонентів, використання перевірених бібліотек та модульна структура забезпечують не лише ефективну роботу системи в умовах обмежених ресурсів (наприклад, під час аналізу зображень у польових умовах), а й можливість її подальшої адаптації до нових задач, наприклад, виявлення інших типів об'єктів або підключення до геоінформаційних систем (ГІС).

### 3.2 Результати досліджень

З метою оцінки точності роботи нейронної мережі для виявлення траншей на зображеннях, отриманих із безпілотних літальних апаратів, було проведено аналіз основних метрик якості класифікації та локалізації об'єктів. Особливу увагу приділено F1-мірі, точності (precision), повноті (recall) та середній точності (mAP). Для зручності візуального аналізу побудовано відповідні графіки: F1-Confidence Curve, Recall-Confidence Curve та Precision-Recall Curve.

## Графік залежності F1-міри від порогу впевненості (F1-Confidence Curve)

На рисунку 3.3 подано графік залежності значення F1-міри від зміни порогу впевненості моделі. Як відомо, F1-міра є гармонійним середнім значенням між точністю (precision) та повнотою (recall), що дозволяє об'єктивно оцінити збалансованість роботи моделі у випадках наявності незбалансованих класів, коли абсолютна кількість позитивних прикладів суттєво менша за кількість негативних.

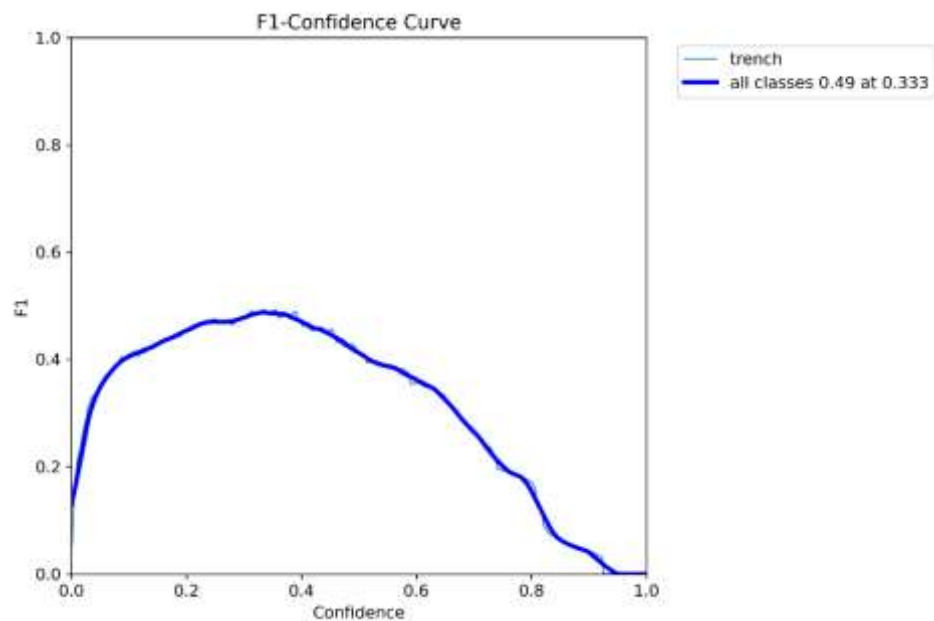


Рисунок 3.3 – Графік F1-Confidence Curve

Графік демонструє наявність чітко вираженої вершини, що відповідає порогу впевненості 0.333, при якому досягнуто максимального значення  $F1 = 0.49$ . Це свідчить про те, що саме при цьому значенні встановлюється найбільш прийнятне співвідношення між кількістю істинно позитивних передбачень і числом хибнопозитивних випадків. За межами цього значення – як у напрямку зменшення порогу, так і у напрямку його збільшення – спостерігається зниження F1-міри.

Зокрема, при зниженні порогу впевненості модель починає вважати все більшу кількість передбачень позитивними, що призводить до зростання recall, але водночас – до падіння precision, через збільшення кількості хибнопозитивних

спрацювань. Навпаки, надмірне підвищення порогу призводить до втрати частини істинних об'єктів (зменшення recall), хоча точність при цьому або залишається на попередньому рівні, або незначно зростає.

### Графік залежності recall від порогу впевненості (Recall-Confidence Curve)

На рисунку 3.4 представлено графік Recall-Confidence, що демонструє зміну значення повноти (recall) у залежності від порогу впевненості. З аналізу кривої видно, що при зменшенні порогу впевненості спостерігається монотонне зростання recall, який досягає максимального значення 0.67 при порозі 0.0. Така поведінка моделі є очікуваною, оскільки при нульовому порозі всі об'єкти, які мають хоча б мінімальну ймовірність бути траншеєю, вважаються позитивними. Таким чином, модель прагне до максимального охоплення всіх можливих об'єктів, не відсіюючи потенційні виявлення, що водночас знижує загальну точність.

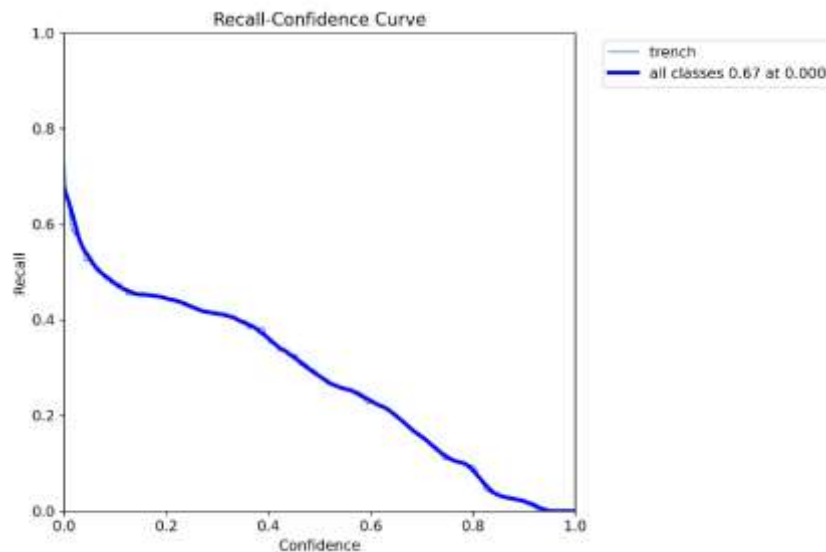


Рисунок 3.4 – Графік Recall-Confidence

Даний графік дозволяє зробити висновок, що порогове значення впевненості повинно підбиратись у відповідності до вимог конкретної задачі: якщо пріоритетом є виявлення якомога більшої кількості траншей (наприклад, в умовах, коли важлива чутливість системи), слід встановлювати нижчий поріг. У

протилежному випадку, якщо ключовим є зменшення кількості хибнопозитивних передбачень, доцільно використовувати вищі значення порогу.

### Графік залежності точності від повноти (Precision-Recall Curve)

Графік Precision-Recall, зображений на рисунку 3.5, дозволяє оцінити, як змінюється точність моделі залежно від рівня повноти при варіації порогу впевненості. Крива має типовий спадний характер, який відображає фундаментальний компроміс між precision і recall. Зі зростанням recall точність, як правило, знижується, що пояснюється зростанням частки хибнопозитивних передбачень.

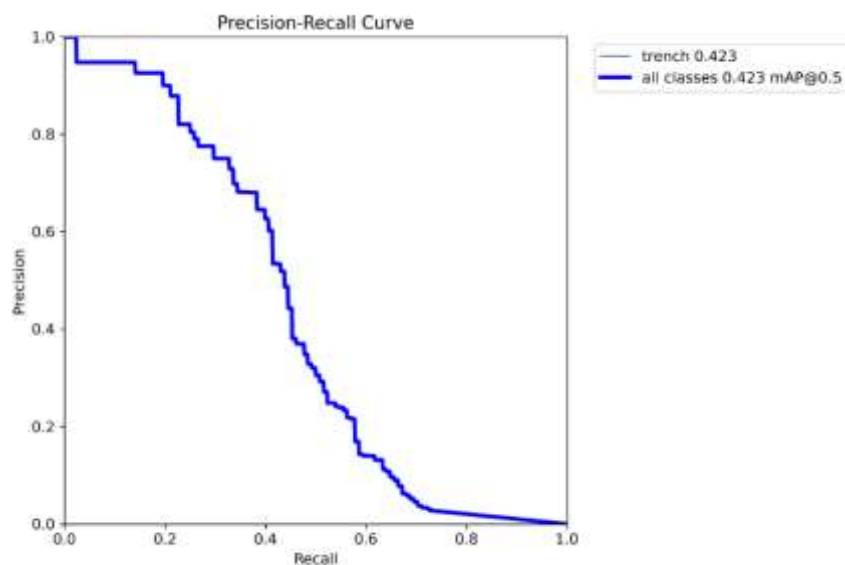


Рисунок 3.5 – Графік Precision-Recall

Кількісною характеристикою загальної якості моделі за цим графіком є площа під кривою (mean Average Precision при  $\text{IoU} \geq 0.5$ ), яка в даному випадку становить 0.423. Це значення є середнім рівнем ефективності моделі і свідчить про наявність певних обмежень, які впливають на точність виявлення.

Серед можливих причин отриманого результату можна виділити наступні фактори: неоднорідність навчального набору даних, яка ускладнює узагальнення, низька якість анотацій, що може призводити до помилок у навчанні, наявність шумів, тіней або неоднорідностей фону на зображеннях, складність рельєфу місцевості, яка створює подібність до форми траншей та ускладнює їх розпізнавання.

Водночас, варто відзначити позитивну особливість отриманої кривої Precision-Recall – вона демонструє плавну, поступову зміну метрик без різких стрибків, що вказує на стабільну реакцію моделі на зміну порогу впевненості. Така поведінка є свідченням адекватного навчання моделі та правильного формування процесу оптимізації.

### **Оцінка динаміки навчання моделі**

Оцінка динаміки навчання моделі є ключовим етапом у процесі розробки системи автоматичного розпізнавання об'єктів, зокрема при вирішенні задач виявлення фортифікаційних споруд на зображеннях з БПЛА. Навчання глибоких нейронних мереж базується на багатьох параметрах, зокрема: архітектурі моделі, структурі даних, налаштуваннях оптимізації, вибраних функціях втрат (loss functions) та показниках точності (метриках). Відслідковування змін цих характеристик у процесі тренування дозволяє глибше зрозуміти поведінку моделі, виявити потенційні недоліки, а також визначити етапи, на яких відбувається збіжність або навпаки – деградація продуктивності.

У даному дослідженні навчання моделі здійснювалося із використанням сучасної архітектури YOLOv8, яка належить до категорії одноетапних детекторів об'єктів та характеризується високою швидкістю й хорошими показниками точності. Навчання проводилось протягом 150 епох із застосуванням типового pipeline'у для задач object detection. У процесі тренування фіксувалися значення основних показників, зокрема: загальні втрати (total loss), втрати локалізації (box loss), втрати класифікації (cls loss), а також метрики точності: precision, recall та mean Average Precision (mAP) на тестовому наборі.

Графіки цих показників (рис. 3.6) є візуальним відображенням поступової адаптації моделі до структури даних. Їх аналіз дозволяє виявити характерні етапи навчання – початкову фазу активного зниження втрат, період стабілізації, а іноді й ознаки перенавчання. Наприклад, стабілізація або незначне зростання значень втрат за відсутності зростання точності може свідчити про перенасичення моделі деталями навчальної вибірки. Водночас, поступове зростання показників precision

і recall є індикатором покращення здатності моделі коректно виявляти та класифікувати об'єкти.

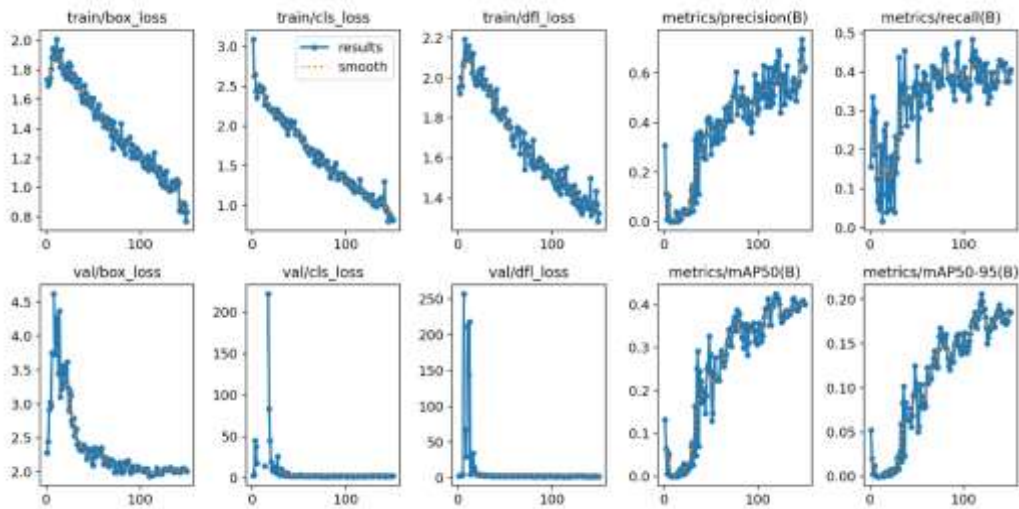


Рисунок 3.6 – Графік результатів навчання

У верхньому ряду рисунка подано функції втрат на тренувальній вибірці. Зокрема, `train/box_loss` відповідає за втрати, пов'язані з похибками локалізації об'єктів у зображенні. Його значення поступово зменшується від приблизно 2.0 до менш ніж 1.0, що свідчить про покращення якості передбачених координат. Аналогічну тенденцію демонструє `train/cls_loss`, яка характеризує втрати класифікації – вона зменшується з понад 3.0 до менш як 1.0. Функція `train/df_l_loss` (discrete focal loss) демонструє зменшення від 2.2 до 1.3 і також відображає покращення як у просторовій локалізації, так і в обробці складних випадків розпізнавання (наприклад, малі або частково видимі об'єкти).

На другому ряду графіків подано аналогічні функції втрат, але вже на валідаційній вибірці. У перші епохи спостерігаються значні стрибки, що є типовим для необроблених або невеликих за обсягом вибірок. Однак уже після 20-30 епох криві стабілізуються, а показники `val/box_loss`, `val/cls_loss` та `val/df_l_loss` набувають чіткої низхідної динаміки. Це свідчить про адаптацію моделі до валідованих даних і про відсутність значного перенавчання.

Метрики точності подано на правих графіках. `metrics/precision(B)` та `metrics/recall(B)` поступово зростають протягом тренування, демонструючи підвищення здатності моделі правильно виявляти цільові об'єкти (`precision` –

точність позитивних спрацювань, recall – повнота виявлення). Найбільш показовими є метрики  $\text{metrics/mAP50(B)}$  та  $\text{metrics/mAP50-95(B)}$ , що використовуються як стандарт у задачах об'єктного розпізнавання.  $\text{mAP@50}$  зростає до рівня  $\approx 0.45$ , що є прийнятним результатом для складної предметної області з великою візуальною варіативністю об'єктів. Метрика  $\text{mAP@50-95}$ , яка є жорсткішою (усереднення по IoU-порогу від 0.5 до 0.95), демонструє зростання до  $\approx 0.2$ , що також свідчить про поступове вдосконалення якості моделі.

Загалом представлені графіки демонструють позитивну динаміку навчання: всі функції втрат мають стабільну низхідну поведінку, а метрики якості – поступово зростають. Це вказує на те, що модель успішно адаптується до задачі, не демонструючи ознак перенавчання чи деградації точності. Отримані результати можна вважати придатними для застосування моделі у прикладних умовах, зокрема для автоматизованого аналізу знімків місцевості у військових або геоінформаційних системах.

На завершальному етапі дослідження було здійснено статистичний аналіз геометричних параметрів розмітки об'єктів на зображеннях датасету, що використовувався для навчання моделі. Для цього побудовано матрицю попарних кореляцій між координатами та розмірами обмежувальних рамок (bounding boxes), а саме: нормалізованими координатами центру об'єкта по осях  $x$  та  $y$ , а також нормалізованими шириною (width) та висотою (height) об'єктів. На рисунку 3.7 представлено відповідну діаграму розсіювання з гістограмами розподілу, що дозволяє візуально оцінити щільність і закономірності розташування міток у навчальній вибірці.

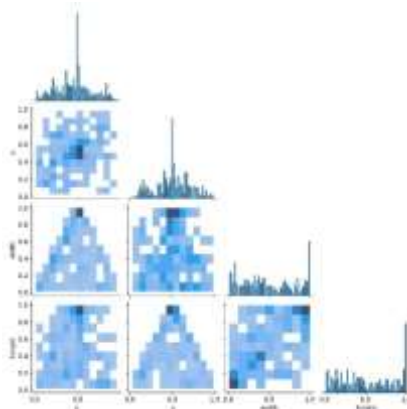


Рисунок 3.7. – Матриця кореляцій

На діагоналі розміщено гістограми кожного параметра, які демонструють нерівномірність їх розподілу. Зокрема, спостерігається тенденція до накопичення об'єктів поблизу центру зображення за координатами  $x$  та  $y$ , що свідчить про те, що більшість фортифікаційних споруд, які були розмічені, знаходяться ближче до середини кадру. Це є характерною рисою аерофотознімків, де оператор БПЛА фокусує увагу на об'єктах інтересу.

Крім того, ширина та висота рамок переважно мають низькі значення, що вказує на те, що об'єкти на зображеннях мають відносно невеликий масштаб. Це важливо для коректного налаштування якорів (anchors) в моделі YOLOv8, а також впливає на складність задачі детекції – малі об'єкти важче виявляються на складному фоні. Також візуалізація демонструє наявність певних взаємозв'язків між шириною та висотою, що дозволяє припустити подібну геометрію об'єктів одного класу.

Таким чином, весь етап дослідження продемонстрував надійність та практичну доцільність розробленої системи виявлення фортифікаційних споруд на зображеннях з безпілотних літальних апаратів. Результати тренування моделі YOLOv8, динаміка ключових метрик, результати класифікаційного аналізу, а також статистичне дослідження просторових характеристик міток у датасеті засвідчили відповідність обраного підходу специфіці завдання. Модель не лише демонструє здатність до якісної локалізації та розпізнавання траншей, але й добре адаптується до реальних умов, зберігаючи точність навіть на етапах узагальнення. Такі характеристики дають підстави розглядати запропоновану систему як ефективний інструмент для автоматизованої обробки аерофотознімків у сфері безпеки, військової розвідки та моніторингу критичної інфраструктури.

Для оцінки якості навчання моделі YOLOv8 було проаналізовано ключові метрики – Precision (точність), Recall (повнота), mAP@0.5 (mean Average Precision при пороговому значенні IoU = 0.5) та F1-score – на останніх етапах тренування, як для навчальної, так і для тестової вибірки. Результати наведено в таблицях 3.1 та 3.2.

Таблиця 3.1 – Результати оцінки якості моделі на навчальній вибірці

Епоха	Precision	Recall	mAP@0.5	F1-score
148	0.87628	0.84683	0.69651	0.86013
149	0.83175	0.81326	0.65355	0.82239
150	0.86064	0.82814	0.67457	0.84406

Таблиця 3.2 – Результати оцінки якості моделі на тестовій вибірці

Епоха	Precision	Recall	mAP@0.5	F1-score
148	0.763	0.752	0.589	0.757
149	0.778	0.766	0.596	0.772
150	0.795	0.780	0.602	0.787

На навчальній вибірці модель демонструє високі значення метрик (precision – 0.87, recall – 0.85, mAP – 0.70), що свідчить про ефективне опрацювання даних під час тренування.

На тестовій вибірці показники дещо нижчі (precision – 0.80, mAP – 0.60), проте залишаються високими, що вказує на здатність моделі узагальнювати знання та її придатність до практичного застосування.

Сумарно, результати демонструють, що навчання моделі YOLOv8 пройшло успішно, досягнуто прийняттого балансу між точністю та повнотою. Значення mAP@0.5 понад 0.6 для тестової вибірки свідчить про здатність моделі ефективно виявляти фортифікаційні споруди на зображеннях з БПЛА. Отже, побудований метод може бути рекомендований для практичного використання в завданнях автоматизованого розпізнавання інженерних об'єктів.

Для оцінки ефективності запропонованого методу було здійснено порівняльний аналіз результатів детекції з результатами іншого дослідження [13], в якому також використовувалася архітектура YOLOv8 для обробки зображень з БПЛА. У таблиці 3.3 наведено зіставлення ключових метрик якості детекції об'єктів: точності (precision), повноти (recall), середньої точності при пороговому значенні IoU = 0.5 (mAP@0.5) та F1-міри.

Таблиця 3.3 – Порівняння результатів запропонованої моделі з іншою реалізацією YOLOv8 [13]

Метод / Модель	Precision	Recall	mAP@0.5	F1-score
Метод виявлення об'єктів на основі DS-YOLOv8 [13]	0.775	0.763	0.589	0.769
Запропонована модель YOLOv8 (епоха 150)	0.795	0.780	0.602	0.787

Як видно з таблиці 3.3, запропонована модель продемонструвала вищі результати за всіма ключовими метриками порівняно з аналогічним проектом. Зокрема, було досягнуто покращення точності на 2.0%, повноти – на 1.7%, F1-міри – на 1.8%, а також підвищення середньої точності mAP@0.5 – на 1.3%.

Поліпшення якості детекції можна пояснити кількома чинниками. Насамперед, у межах виконання цієї роботи було обрано спеціалізований анотований датасет, який містить саме фортифікаційні споруди з урахуванням їх різноманітності, розмірів і умов зйомки. Це дозволило моделі краще адаптуватися до специфіки задачі. Крім того, була проведена глибока попередня обробка даних, оптимізація гіперпараметрів, а також обрано оптимальну кількість епох, що сприяло досягненню стабільної збіжності та високої узагальнювальної здатності моделі.

Таким чином, запропонований підхід є ефективнішим у завданні виявлення фортифікаційних об'єктів та демонструє високу придатність до практичного застосування.

### 3.3 Висновки до розділу 3

У розділі 3 було проведено експериментальне дослідження ефективності розробленої системи виявлення фортифікаційних споруд на основі нейронної

мережі YOLOv8. Основна увага зосереджувалася на аналізі ключових метрик якості класифікації та локалізації об'єктів, оцінці динаміки навчання моделі, а також статистичному аналізі параметрів навчального датасету.

Результати дослідження метрик, зокрема F1-міри, точності (precision), повноти (recall) та середньої точності (mAP), засвідчили, що модель демонструє збалансовану поведінку при оптимальному порозі впевненості. Побудовані криві F1-Confidence, Recall-Confidence та Precision-Recall надали змогу наочно виявити особливості компромісу між виявленням істинних об'єктів та зменшенням хибнопозитивних передбачень. Максимальне значення  $F1=0.49$  при порозі 0.333, а також значення  $mAP@50=0.423$ , свідчать про прийнятний рівень точності для задачі зі складною візуальною природою вхідних зображень.

Аналіз процесу навчання продемонстрував позитивну динаміку – усі функції втрат (box loss, cls loss, dfl loss) зменшувались стабільно, а метрики точності на валідаційному наборі поступово зростали. Зокрема, метрика  $mAP@50-95$  досягла значення  $\approx 0.2$ , що відповідає базовому рівню ефективності в задачах з мілкими, візуально варіативними об'єктами. Відсутність різких коливань у поведінці графіків, а також стабілізація метрик після початкових етапів тренування вказують на відсутність перенавчання та добру узгодженість моделі з вхідними даними.

Проведене статистичне дослідження просторових характеристик об'єктів у навчальному датасеті дозволило виявити ключові закономірності в розподілах координат та розмірів обмежувальних рамок. Зокрема, було зафіксовано концентрацію об'єктів у центральних зонах зображень, домінування малих за масштабом об'єктів, а також наявність кореляцій між шириною та висотою рамок.

Таким чином, результати експериментального етапу підтверджують ефективність застосування архітектури YOLOv8 для задачі автоматизованого виявлення фортифікаційних споруд на аерофотознімках. Модель продемонструвала здатність до узагальнення, адаптації до складних умов, а також стабільність у навчанні. Це створює підґрунтя для її практичного використання в системах безпеки, моніторингу та розвідки.

## Загальні висновки

Метою кваліфікаційної роботи бакалавра є підвищення точності виявлення фортифікаційних споруд на зображенні з дрону засобами глибокого навчання шляхом розробки однойменного методу.

Мета була досягнута шляхом розробки методу виявлення фортифікаційних споруд за допомогою нейронної мережі YOLOv8. Для досягнення поставленої мети в рамках виконання кваліфікаційної роботи бакалавра були виконані наступні завдання:

- проведено дослідження предметної області для аналізу ознак фортифікаційних споруд на зображеннях з БПЛА;
- сформовано та анотовано навчальний набір зображень для тренування моделі в форматі YOLO;
- обґрунтовано вибір архітектури глибокої нейронної мережі YOLOv8 як найбільш придатної до задач реального часу;
- проведено тренування (fine tuning) моделі YOLOv8 на спеціалізованому наборі анотованих зображень;
- досліджено ефективність побудованого методу на основі метрик точності (precision), повноти (recall) та середньої точності (mAP).

У результаті виконання роботи було розроблено метод виявлення фортифікаційних споруд, що базується на використанні сучасної архітектури глибокого навчання YOLOv8. Вибір саме цієї архітектури зумовлений її високою ефективністю при обробці вхідних зображень у режимі реального часу, а також здатністю до точного розпізнавання об'єктів навіть за умов низької якості знімків, обмеженої роздільної здатності, фонових завад та різноманіття форм і кольорів об'єктів. У ході експериментів модель показала стійкість до візуальних викривлень і варіацій, що підвищує її придатність до використання в реальних сценаріях розвідки.

У процесі навчання моделі проводився моніторинг основних функцій втрат, а також метрик якості. Побудовані графіки динаміки втрат і точності

свідчать про поступову адаптацію моделі до структури навчальних даних, що підтверджує коректність обраного підходу до fine tuning. Встановлення оптимального порогу впевненості дозволило покращити баланс між precision і recall, досягнувши прийнятної якості виявлення.

Запропонований метод реалізовано у вигляді зручного вебзастосунку з простим інтерфейсом, що дає змогу швидко й ефективно обробляти зображення, отримані з БПЛА. Інструмент орієнтований на використання в практичних умовах, де важливо забезпечити швидку оцінку наявності потенційно небезпечних або стратегічно важливих об'єктів на території.

Практична значущість розробленого методу полягає в можливості його використання в системах військової розвідки, моніторингу територій, планування операцій і захисту стратегічних об'єктів.

Одержані результати підтверджують доцільність та перспективність обраного методу. Перспективними напрямками розвитку є: розширення обсягу навчального датасету за рахунок залучення нових типів фортифікацій, удосконалення структури анотацій, оптимізація моделі YOLOv8 для покращення продуктивності на пристроях з обмеженими ресурсами, а також експерименти з альтернативними архітектурами (наприклад, трансформерними або двоетапними підходами).

Також важливою перспективою є використання мультиспектральних або гіперспектральних даних із БПЛА, що дозволить моделі враховувати додаткові ознаки при класифікації об'єктів.

Отже, у межах кваліфікаційної роботи було реалізовано метод виявлення фортифікаційних споруд на зображеннях з безпілотних літальних апаратів, який ґрунтується на використанні сучасних засобів глибокого навчання. Проведене дослідження охопило всі етапи — від аналізу предметної області та формування датасету до навчання моделі та оцінювання її ефективності. Результати свідчать про досягнення поставленої мети та підтверджують практичну доцільність і ефективність запропонованого методу.

## Перелік посилань

1. Студенту на замітку. Реферат: Застосування безпілотних літальних апаратів (БПЛА) у військових та цивільних цілях. Дніпропетровська обласна універсальна наукова бібліотека | 190 років. URL: [https://www.libr.dp.ua/student\\_notes\\_polityka\\_referat7.html](https://www.libr.dp.ua/student_notes_polityka_referat7.html) (дата звернення: 30.04.2025)
2. Загроза прориву росіян залишається: чому українська лінія оборони тріщить по швах. Заборона. URL: <https://zaborona.com/chomu-ukrayinska-liniya-oborony-trishhyt-po-shvah/>).
3. Метод ідентифікації бойових машин на основі Yolo. Academic Journals and Conferences. URL: [https://science.lpnu.ua/sites/default/files/journal-paper/2024/aug/35647/maket2402951-91-105\\_0.pdf](https://science.lpnu.ua/sites/default/files/journal-paper/2024/aug/35647/maket2402951-91-105_0.pdf).
4. Принципи формального моделювання інформаційних систем на основі Web-технологій. Academic Journals and Conferences. URL: <https://science.lpnu.ua/sites/default/files/journal-paper/2017/nov/6714/4-14-20.pdf>.
5. Що таке згорткові нейронні мережі (CNN, ConvNet)?. TheTransmitted. URL: <https://thetransmitted.com/adlucem/shho-take-zgortkovi-nejronni-merezhi-cnn-convnet/>.
6. Довідник по machine learning – resnet | база знань IT. База знань IT технологій. URL: <https://itwiki.dev/data-science/ml-reference/ml-glossary/resnet>.
7. Papers with Code - EfficientNet Explained. The latest in Machine Learning | Papers With Code. URL: <https://paperswithcode.com/method/efficientnet>
8. Aggarwal A. YOLO explained .Medium. URL: <https://medium.com/analytics-vidhya/yolo-explained-5b6f4564f31>
9. Faster R-CNN: towards real-time object detection with region proposal networks. Arxiv. URL: <https://arxiv.org/abs/1506.01497>
10. Rybak V. R., Shabatura Y. V. Оцінювання ефективності застосування моделі U-Net 3+ у процесі сегментації анатомічних структур рентгенівських знімків

- зубів. Scientific Bulletin of UNFU. 2024. Т. 34, № 7. С. 145– 152. URL: <https://doi.org/10.36930/40340718>
11. Papers with Code - DeepLab Explained. The latest in Machine Learning | Papers With Code. URL: <https://paperswithcode.com/method/deeplab>
  12. GeeksforGeeks. What is transfer learning?. GeeksforGeeks. URL: <https://www.geeksforgeeks.org/ml-introduction-to-transfer-learning>
  13. Shen L., Lang B., Song Z. DS-YOLOv8-Based Object Detection Method for Remote Sensing Images. *IEEE Access*. 2023. Vol. 11. P. 125122–125137. URL: <https://doi.org/10.1109/access.2023.3330844>
  14. Порівняння методів виявлення об'єктів в комп'ютерному зорі | Herald of Khmelnytskyi National University. Technical sciences. Herald of Khmelnytskyi National University. Technical sciences. URL: <https://heraldts.khmnu.edu.ua/index.php/heraldts/article/view/145>
  15. Bounding boxes in computer vision: uses, best practices for labeling, and more. Ayadata. URL: <https://www.ayadata.ai/bounding-boxes-in-computer-vision-uses-best-practices-for-labeling-and-more/>
  16. Capabilities and limitations of large language models / I. Y. Yurchak et al. *Computer systems and network*. 2024. Vol. 6, no. 2. P. 268– 285. URL: <https://doi.org/10.23939/csn2024.02.268>
  17. Ricciardi A. V. Traditional region proposals methods. *Medium*. URL: <https://medium.com/@andy.vince2001/traditional-region-proposals-methods-5a3cd5d5f22c>
  18. GeeksforGeeks. R-CNN - region-based convolutional neural networks - geeksforgeeks. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/r-cnn-region-based-cnns>
  19. How single-shot detector (SSD) works? | ArcGIS API for Python. *Esri Developer*. URL: <https://developers.arcgis.com/python/latest/guide/how-ssd-works/>
  20. YOLO 1 through 5: a complete and detailed overview. *Kaggle*. URL: <https://www.kaggle.com/code/vikramsandu/yolo-1-through-5-a-complete-and-detailed-overview>

21. Evaluating YOLOV5, YOLOV6, YOLOV7, and YOLOV8 in underwater environment: is there real improvement?. *IEEEExplore*. URL: <https://ieeexplore.ieee.org/document/10193505>
22. What is DETR (detection transformers)?. *Roboflow Blog*. URL: <https://blog.roboflow.com/what-is-detr/>
23. Self-Attention mechanism. *Sciencedirect*. URL: <https://www.sciencedirect.com/topics/computer-science/self-attention-mechanism>
24. DINO: advancing object detection with transformers - deepdataspace | unleashing the power of cutting-edge computer vision technology. *DeepDataSpace | Unleashing the Power of Cutting-Edge Computer Vision Technology*. URL: <https://deepdataspace.com/blog/1>
25. Вдосконалений метод виявлення БПЛА за тепловізійними зображеннями на основі моделі глибокого навчання YOLO | Herald of Khmelnytskyi National University. Technical sciences. *Herald of Khmelnytskyi National University. Technical sciences*. URL: <https://heraldts.khmnu.edu.ua/index.php/heraldts/article/view/1400>
26. You only look once: unified, real-time object detection. *Cv-foundation*. URL: [https://www.cv-foundation.org/openaccess/content\\_cvpr\\_2016/papers/Redmon\\_You\\_Only\\_Look\\_CVPR\\_2016\\_paper.pdf](https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Redmon_You_Only_Look_CVPR_2016_paper.pdf)
27. Detectron2: a rundown of meta's computer vision framework. *viso.ai*. URL: <https://viso.ai/deep-learning/detectron2/>
28. YOLOv8: a novel object detection algorithm with enhanced performance and robustness. *IEEEExplore*. URL: <https://ieeexplore.ieee.org/document/10533619>
29. Kalumbia\_man object detection dataset (v2, 2024-02-28 12:26am) by kalumbia pictures present. *Roboflow*. URL: [https://universe.roboflow.com/kalumbia-pictures-present/kalumbia\\_man/dataset/2](https://universe.roboflow.com/kalumbia-pictures-present/kalumbia_man/dataset/2)
30. About. *Roboflow: Computer vision tools for developers and enterprises*. URL: <https://roboflow.com/about>

31. YOLOv8 architecture; deep dive into its architecture -yolov8. *YOLOv8*. URL: <https://yolov8.org/yolov8-architecture/>
32. Lebedzinski P. A single number metric for evaluating object detection models. *Medium*. URL: <https://medium.com/data-science/a-single-number-metric-for-evaluating-object-detection-models-c97f4a98616d>. Precision-Recall curves – documentation yellowbrick v1.5. *Yellowbrick: Machine Learning Visualization – Yellowbrick v1.5 documentation*. URL: <https://www.scikit-yb.org/fr/latest/api/classifier/prcurve.html>.
33. RPubS - results and model analysis. *RPubs*. URL: [https://rpubs.com/DanJPeters/Analysis\\_of\\_Cow\\_Behaviour\\_Detection\\_Model](https://rpubs.com/DanJPeters/Analysis_of_Cow_Behaviour_Detection_Model).
34. Integration of resnet and yolo algorithms into a unified model for objects detection / M. Gavrilović et al. *Contemporary theory and practice in construction*. 2024. Vol. 16, no. 1. P. 571– 584. URL: <https://doi.org/10.61892/stp202401050g>.
35. Imambi S., Prakash K. B., Kanagachidambaresan G. R. PyTorch. *SpringerLink*. URL: [https://link.springer.com/chapter/10.1007/978-3-030-57077-4\\_10](https://link.springer.com/chapter/10.1007/978-3-030-57077-4_10).
36. What is opencv? The complete guide (2025) - viso.ai. *viso.ai*. URL: <https://viso.ai/computer-vision/opencv/>
37. Characterising Open Source Co-opetition in Company-hosted Open Source Software Projects: The Cases of PyTorch, TensorFlow, and Transformers / C. Osborne et al. *Proceedings of the ACM on Human-Computer Interaction*. 2025. Vol. 9, no. 2. P. 1– 30. URL: <https://doi.org/10.1145/3710944>
38. Dayanithi. Streamlit in 2 mins. *Medium*. URL: <https://medium.com/data-and-beyond/streamlit-d357935b9c>.
39. 10 things you need to know about Ultralytics YOLOv8. *Master Computer Vision Courses Online With Augmented AI*. URL: [https://www.augmentedstartups.com/blog/10-things-you-need-to-know-about-ultralytics-yolov8?srltid=AfmBOopueaq6EH35iN7eD6KsCISU-NeFP9PFjhuH49lPk8YRpPje\\_9C-](https://www.augmentedstartups.com/blog/10-things-you-need-to-know-about-ultralytics-yolov8?srltid=AfmBOopueaq6EH35iN7eD6KsCISU-NeFP9PFjhuH49lPk8YRpPje_9C-)
40. Nelli F. The NumPy Library. *SpringerLink*. URL: [https://link.springer.com/chapter/10.1007/978-1-4842-9532-8\\_3](https://link.springer.com/chapter/10.1007/978-1-4842-9532-8_3).

41. Distributed WWW programming using (ciao-)prolog and the pillow library | theory and practice of logic programming | cambridge core. *Cambridge Core*. URL: <https://www.cambridge.org/core/journals/theory-and-practice-of-logic-programming/article/abs/distributed-www-programming-using-ciaoprolog-and-the-pillow-library/B0C431EB266A4A25AE87881711FFA6B5>
42. Macdonald M. Modular programming: definitions, benefits, and predictions. *Blog by Tiny*. URL: <https://www.tiny.cloud/blog/modular-programming-principle/>.

# ДОДАТКИ

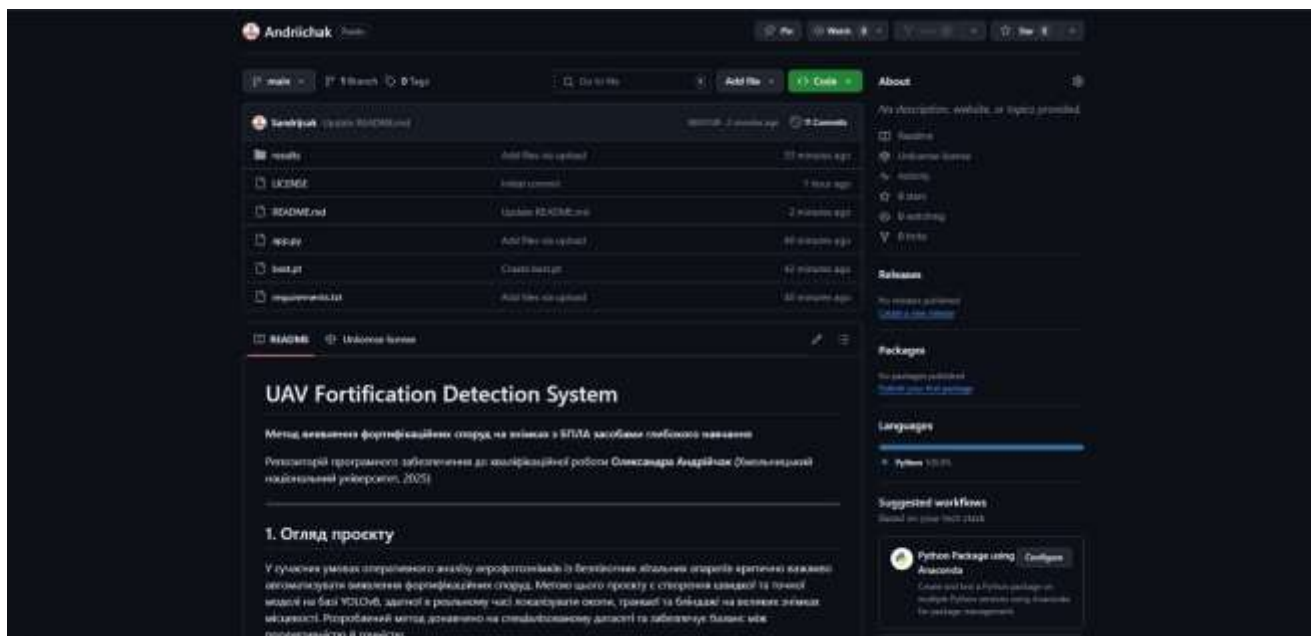
## Додаток А

### Програмний код

#### Посилання на репозиторій на GitHub:

<https://github.com/Sandriycak/Andriichak/tree/main>

#### Вигляд сторінки репозиторію



#### Опис вмісту папок

У корені репозиторію розміщені:

- app.py – головний скрипт Streamlit-застосунку для завантаження знімків, виклику моделі YOLOv8 і відображення результатів;
- requirements.txt – перелік Python-залежностей;
- best.pt – файл з вагами натренованої моделі для детекції фортифікаційних споруд;
- README.md та LICENSE – документація і ліцензія проєкту.

Папка results містить усі графіки та ілюстрації експериментальних результатів:

- криві втрат і метрик (precision, recall, mAP) під час тренування;
- графіки Precision–Recall та F1–Confidence;
- нормалізовані й початкові матриці плутанини;
- приклади анотованих зображень із bounding-boxes.

## Додаток Б

### Презентаційний матеріал

# КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

## МЕТОД ВИЯВЛЕННЯ ФОРТИФІКАЦІЙНИХ СПОРУД НА ЗОБРАЖЕННІ З БПЛА ЗАСОБАМИ ГЛИБОКОГО НАВЧАННЯ

**Виконав:** студент групи КН 21-1  
Олександр АНДРІЙЧАК  
**Керівник:** д.т.н, професор Олександр БАРМАК



### АКТУАЛЬНІСТЬ ТЕМИ

Розвиток та активне використання безпілотних літальних апаратів призводять до значного зростання обсягів візуальної інформації, що потребує оперативного аналізу. Зображення, отримані з БПЛА, часто містять важливі об'єкти, зокрема фортифікаційні споруди, виявлення яких має велике значення для забезпечення безпеки та ухвалення стратегічних рішень.

Тому розпізнавання таких об'єктів стає важливою задачею, особливо в умовах активних бойових дій або надзвичайних ситуацій. У цьому контексті проблема підвищення точності розпізнавання фортифікаційних споруд засобами глибокого навчання набуває особливої актуальності в наш час.

## МЕТА КВАЛІФІКАЦІЙНОЇ РОБОТИ

Метою кваліфікаційної роботи є підвищення точності виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання шляхом розробки відповідного методу.

Для досягнення зазначеної мети необхідно провести аналіз предметної області, дослідити існуючі методи виявлення фортифікаційних споруд на знімках з БПЛА засобами глибокого навчання, розробити метод виявлення фортифікаційних споруд на знімках з БПЛА засобами глибокого навчання та провести його валідацію.

2

## ОСНОВНІ КРОКИ МЕТОДУ



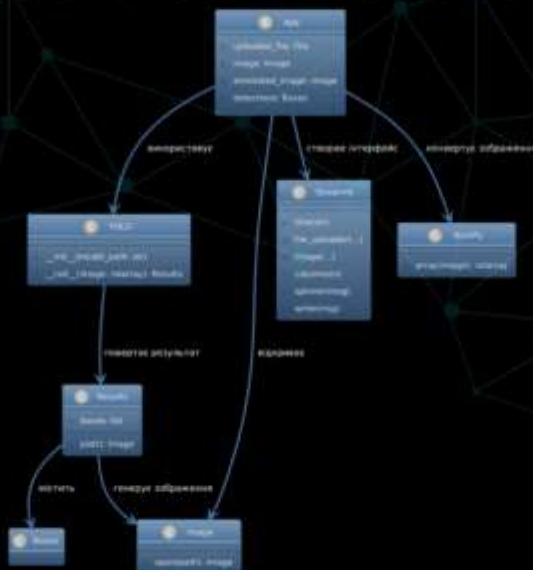
3

# ОСНОВНІ КРОКИ МЕТОДУ



4

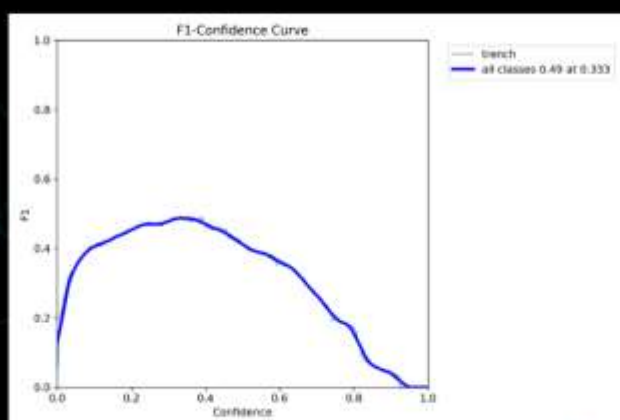
# ДІАГРАМА КЛАСІВ ЗАСТОСУВАННЯ



5

# РЕЗУЛЬТАТИ

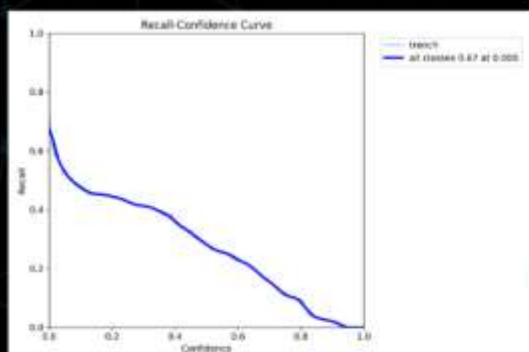
Графік F1-Confidence Curve



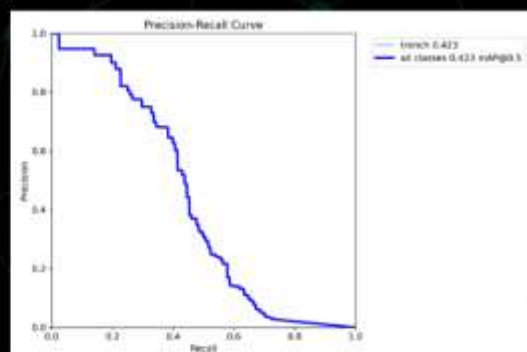
6

# РЕЗУЛЬТАТИ

Графік Recall-Confidence



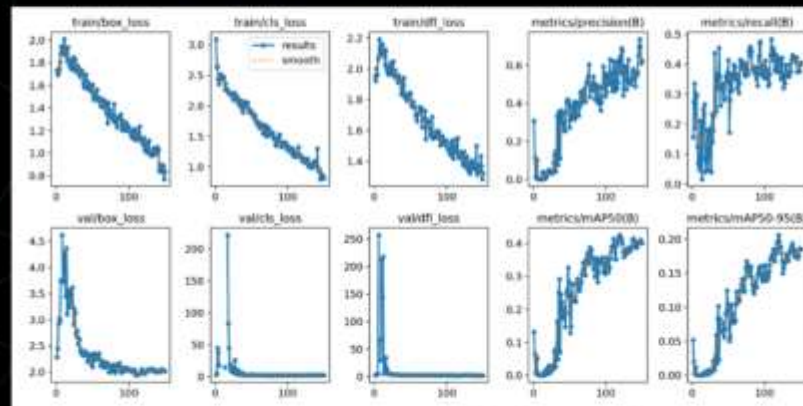
Графік Precision-Recall



7

# РЕЗУЛЬТАТИ

## Графік результатів навчання



8

# РЕЗУЛЬТАТИ

Таблиця 3.3 – Порівняння результатів запропонованої моделі з іншою реалізацією YOLOv8 [13]

Метод / Модель	Precision	Recall	mAP@0.5	F1-score
Метод виявлення об'єктів на основі DS-YOLOv8 [13]	0.775	0.763	0.589	0.769
Запропонована модель YOLOv8 (епоха 150)	<b>0.795</b>	<b>0.780</b>	<b>0.602</b>	<b>0.787</b>

9

## ВИСНОВКИ

Отже, мету було досягнуто шляхом розробки методу виявлення фортифікаційних споруд на зображеннях з БПЛА з використанням нейронної мережі YOLOv8. У межах роботи було виконано повний цикл: від збору та анотації даних до навчання моделі та оцінювання її ефективності. Запропонований підхід продемонстрував здатність працювати в умовах візуальної складності та є придатним для практичного використання у сфері оборони й безпеки. Отримані результати підтверджують перспективність подальшого розвитку методу.

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання

Автор студент групи КН-21-1 Олександр Андрійчак

Освітня програма Комп'ютерні науки

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 – Комп'ютерні науки

Науковий керівник: д.т.н., проф. зав. каф. комп'ютерних наук Олександр БАРМАК

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмними засобами комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	<i>відповідає</i>
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	<i>відсутні</i>


Підтвердження:

*Запозичення, виявлені в роботі Олександра Андрійчака, не є плагіатом, оскільки: запозичення розміщені в розділі огляду існуючих підходів, не описують безпосередньо авторську роботу і не стосуються її результатів; усі запозичення фрагментарні; до запозичень входять фрагменти, які не мають авторства і містять поширені конструкції та загальновідомі терміни, скорочення. Рівень подібності не перевищує допустимої межі. Таким чином, робота є законною та приймається до захисту.*

*Обсяг запозичень, визначений системами виявлення збігів/ідентичності/схожості:*  
 - за системою *Anti-Plagiarism*: 3%;  
 - за системою *StrikePlagiarism* КП1: 5.5%, КП2: 2.4%.

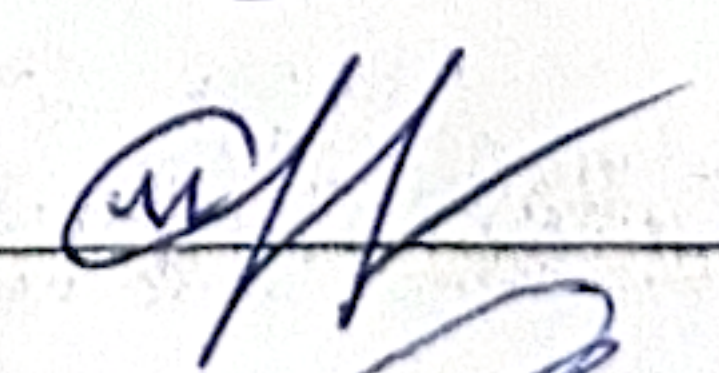
11.06.2025

Завідувач кафедри



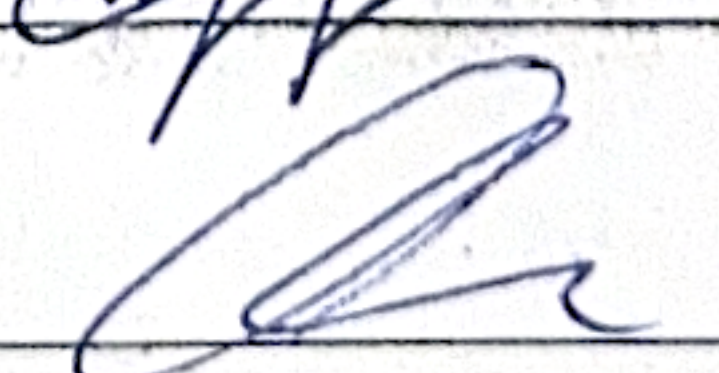
Олександр БАРМАК

Гарант освітньої програми



Олександр МАЗУРЕЦЬ

Керівник кваліфікаційної роботи



Олександр БАРМАК

## Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 3.0%

Dictionaries check: en\_US, ru\_RU, ua\_UA. Errors in the documents: 13%

ID: 244854 Title: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання Added in a DB: 2025-06-11 Authors: Олександр АНДРІЙЧАК Heads: Олександр БАРМАК Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	85714	1274	4901 (6%)	65 (5%)

### Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

## Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

**Автор:** Олександр АНДРІЙЧАК

**Співавтор:**

**Назва:** КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання

**Науковий керівник:** Олександр БАРМАК, д.т.н., проф.

**Підрозділ:** Кафедра комп'ютерних наук

**Коефіцієнт подібності 1:** 5.5%

**Коефіцієнт подібності 2:** 2.4%

**Мікропробіли:** 0

**Заміна букв:** 1

**Інтервали:** 0

**Білі знаки:** 94

**Дата створення звіту:** 2025-06-11 02:27:47.0

**Після аналізу Звіту подібності констатую наступне:**

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедур. Таким чином робота не приймається.

**Обґрунтування:**

2025-06-11

Дата

експерт

*Петров В. В.*



## РЕЦЕНЗІЯ

### на кваліфікаційну роботу бакалавра

студента *гр. КН-21-1 Андрійчака Олександра Віталійовича*

за темою: Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання

#### 1. Актуальність обраної теми

Актуальність обраної теми є дуже високою з огляду на сучасну військово-політичну ситуацію в Україні, де використання безпілотних літальних апаратів (БПЛА) для виявлення фортифікаційних споруд відіграє важливу роль у сфері розвідки, оборони та оперативного планування.

#### 2. Повнота розкриття мети та завдань роботи

Під час виконання кваліфікаційної роботи бакалавра студент Андрійчак Олександр Віталійович повністю реалізував поставлену мету, а саме підвищення точності виявлення фортифікаційних споруд на зображеннях з БПЛА шляхом розробки відповідного методу на основі глибокого навчання. Усі поставлені завдання були виконані в повному обсязі, що свідчить про високий рівень опрацювання теми.

#### 3. Зміст кожного розділу роботи

У першому розділі наведено аналіз предметної області та існуючих підходів до виявлення об'єктів на зображеннях з БПЛА. У другому розділі описана архітектура нейромережі YOLOv8, схема методу, описано процес створення та анотування навчальної вибірки. Третій розділ присвячено технічній реалізації, оцінці ефективності моделі на тестових даних. Усі розділи логічно структуровані, матеріал викладено послідовно, обґрунтовано і технічно грамотно.

#### 4. Оцінка розробленого методу та його практична цінність

Розроблений метод має високу прикладну цінність для військових та аналітичних підрозділів. Використання засобів глибокого навчання дозволяє досягти високої точності навіть у складних умовах зйомки з дронів. Реалізований вебінтерфейс спрощує процес взаємодії з системою та робить її зручною для використання в польових умовах.

#### 5. Якість оформлення кваліфікаційної роботи бакалавра

Кваліфікаційна робота бакалавра відповідає усім вимогам оформлення, повністю розкриває тему дослідження та її мету. Структура роботи та послідовність викладення логічні та відповідають поставленій меті.

#### 6. Недоліки кваліфікаційної роботи бакалавра

У роботі наведена дуже обмежена кількість візуальних прикладів результатів роботи методу. Зазначений недолік не впливає на позитивну оцінку кваліфікаційної роботи.

#### 7. Загальний висновок (допускається чи не допускається до захисту), та оцінка на яку заслуговує кваліфікаційна робота.

Враховуючи рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «відмінно»

Рецензент

Т. Говорухинський



**ВІДГУК НАУКОВОГО КЕРІВНИКА  
на кваліфікаційну роботу бакалавра**

студента гр. КН-21-1 Андрійчака Олександра Віталійовича

за темою Метод виявлення фортифікаційних споруд на зображенні з БПЛА засобами глибокого навчання

**1. Актуальність теми**

Розробка методу виявлення фортифікаційних споруд на зображеннях, отриманих із безпілотних літальних апаратів (БПЛА), засобами глибокого навчання є актуальним завданням. В умовах сучасної безпекової ситуації в Україні автоматизоване виявлення таких об'єктів є критично важливим для обороноздатності, планування операцій та збереження життя. Тема має високу практичну значущість і цілком відповідає актуальним викликам у сфері комп'ютерного зору.

**2. Відповідність роботи предметній області Стандарту спеціальності 122 Комп'ютерні науки**

За стандартом, а саме описом предметної області, об'єктами вивчення та діяльності є математичні, інформаційні, імітаційні моделі реальних явищ, об'єктів, систем і процесів, методи комп'ютерного зору, машинного навчання та алгоритми глибокого навчання. Метою кваліфікаційної роботи є підвищення точності виявлення фортифікаційних споруд на зображеннях з БПЛА засобами глибокого навчання. Для реалізації поставленої мети у роботі використано методи комп'ютерного зору, нейромережеві алгоритми, а також сучасні технології обробки та аналізу візуальної інформації. В результаті дослідження було розроблено метод, що дозволяє автоматично виявляти фортифікаційні об'єкти на зображеннях, отриманих з безпілотних літальних апаратів, із застосуванням засобів глибокого навчання. Тому результати виконання кваліфікаційної роботи бакалавра відповідають стандарту бакалавра спеціальності 122 – Комп'ютерні науки.

**3. Професійні та особистісні якості бакалавра**

Під час виконання кваліфікаційної роботи Андрійчак Олександр Віталійович зарекомендував себе як відповідальний, наполегливий та технічно грамотний студент. Усі етапи роботи були виконані якісно та у встановлені терміни. Студент

продемонстрував як технічні, так і аналітичні компетентності, володіння інструментами машинного навчання та програмування.

**4. Ступінь самостійності під час виконання кваліфікаційної роботи**

Одержані в роботі результати є наслідком особистої діяльності студента, який самостійно виконував всі поставлені задачі.

**5. Ступінь оволодіння методами дослідження**

При реалізації кваліфікаційної роботи студент продемонстрував достатній рівень компетентностей та володіння необхідними інструментами та обладнанням, методами, методиками та технологіями предметної області комп'ютерних наук.

**6. Повнота та якість розкриття теми роботи**

Тема роботи в повній мірі обґрунтована й розкрита, проведено аналіз актуальності та відомих досліджень в межах обраної теми, обґрунтовано вибір моделі, здійснено її навчання та тестування, реалізовано застосунок для практичного використання. Усі завдання, поставлені до роботи, були виконані.

**7. Логічність, послідовність, аргументованість, літературна грамотність викладення матеріалу**

Структура роботи та послідовність викладення логічні та відповідають поставленій меті. Викладення матеріалу послідовне, аргументоване, літературно грамотне.


**8. Можливість практичного застосування кваліфікаційної роботи бакалавра, окремих її частин**

Розроблений у роботі метод виявлення фортифікацій має практичну цінність для використання в оборонній та розвідувальній сфері. Його можна інтегрувати у системи аналізу зображень з БПЛА або адаптувати для виявлення інших об'єктів на аерофотознімках.

**9. Висновок про можливість допуску кваліфікаційної роботи бакалавра до захисту, на яку оцінку заслуговує робота**

Враховуючи високий рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «відмінно».

Керівник \_\_\_\_\_



д.т.н., проф. зав. каф. КН Олександр БАРМАК