

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій і технічного захисту інформації
(повна назва)

Кафедра Радіотехнологій інформаційно-комунікаційних систем
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)

«Проектування інформаційної системи для аналізу повітряних об'єктів з використанням машинного навчання»
(тема)

Виконав:

студент 2 курсу, групи АПСм-22-1

Шендрик О.М.

(прізвище, ініціали)

Спеціальність 126 Інформаційні системи та технології

(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Архітектурне проектування інформаційних систем

(повна назва освітньої програми)

Керівник доц Сайківська Л.Ф.

(посада, прізвище, ініціали)

Допускається до захисту

В.о. зав. кафедри

(підпис)

Зарудний О.А.

(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій і технічного захисту інформації

Кафедра Радіотехнологій інформаційно-комунікаційних систем

Рівень вищої освіти другий (магістерський)

Спеціальність 126 Інформаційні системи та технології

(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Архітектурне проєктування інформаційних систем

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«___» _____ 2023 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Шендрику Олексію Миколайовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Проектування інформаційної системи для аналізу повітряних об'єктів з використанням машинного навчання

затверджена наказом університету від 03.11.2023 р. № 1295 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10. 01. 2023 р.

3. Вихідні дані до роботи Провести аналіз можливостей машинного навчання для задачі виявлення повітряних об'єктів, виконати проєктування технічної архітектури інформаційної системи з використанням можливостей машинного навчання; розробити архітектуру бізнес-моделі інформаційної системи для аналізу повітряних об'єктів.

4. Перелік питань, що потрібно опрацювати в роботі Вступ.

1. Опис базових понять для аналізу об'єктів.

2. Огляд та аналіз існуючих рішень.

3. Розробка технічної архітектури ІС та дослідження інструментаріїв для її реалізації.

4. Розробка бізнес-архітектури ІС для аналізу повітряних об'єктів.

Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Слайди презентації

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	доц. Сайківська Л.Ф		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Огляд літератури за тематикою роботи	05.11.23	вик.
2	Аналіз інструментаріїв розробки	15.11.23	вик.
3	Аналіз інструментаріїв машинного навчання	25.11.23	вик.
4	Розгортання машинного навчання локально	03.12.23	вик.
5	Розробка архітектури ІС	20.11.23	вик.
6	Реалізація архітектурних схем	30.12.23	вик.
7	Оформлення пояснювальної записки	10.01.24	вик.
8	Подання на кафедру	10.01.24	вик.

Дата видачі завдання 03 листопада 2023 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

доц. Сайківська Л.Ф.
(посада, прізвище, ініціали)

РЕФЕРАТ

Кваліфікаційна робота бакалавра складається з пояснювальної записки, що містить 63 сторінок тексту, 37 рисунків, 1 таблицю 32 джерел посилання та 3 додатка.

AI, ML, API, ANALYTICS, MILITARY

Тема кваліфікаційної роботи магістра присвячена документації архітектури ІС, призначеної для використання в літальних пристроях як додатку для детекції та захоплення фокусу на повітряних або наземних цілях.

ABSTRACT

The qualification work of the bachelor consists of an explanatory note containing 63 pages of text, 37 figures, 39 references.

AI. ML. API. ANALYTICS. MILITARY.

The purpose of this qualifying work dedicated to the documentation of IS architecture, intended for use in lethal devices as an additive for detection and storage of focus on surface and ground targets.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	7
ВСТУП.....	8
1 ОПИС БАЗОВИХ ПОНЯТЬ ДЛЯ АНАЛІЗУ ОБ'ЄКТІВ.....	9
1.1 Поняття цифрового аналізу зображень.....	9
1.2 Технічна розуміння аналізу зображень.....	12
1.2.1 Колірна модель.....	12
1.2.2 Ізоляція об'єктів за допомогою порогів.....	15
1.2.3 Розпізнавання об'єкту за допомогою морфологічних операцій.....	17
1.3 Класифікація зображень.....	20
2 ОГЛЯД ТА АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ.....	22
2.1 Використання машинного навчання у військовій справі.....	22
2.2 Використання аналізу зображень у військовій справі.....	24
2.3 Огляд сервісів, що надають послуги ML/AI.....	25
3 РОЗРОБКА ТЕХНІЧНОЇ АРХІТЕКТУРИ ІС ТА ДОСЛІДЖЕННЯ ІНСТРУМЕНТАРІЇВ ДЛЯ ЇЇ РЕАЛІЗАЦІЇ	27
3.1 Розробка функціональних вимог до проектуємої інформаційної системи.....	27
3.2 Архітектурний концепт ІС для аналізу повітряних об'єктів.....	29
3.3 Огляд можливостей ChatGPT для реалізації задачі аналізу повітряних об'єктів.....	30
3.4 Огляд бібліотек моделей навчання для вирішення задачі аналізу повітряних об'єктів.....	32
3.5 Вибір набору даних для реалізації задачі аналізу повітряних об'єктів....	34
3.6 Організація робочого середовища.....	36
3.7 Реалізація навчання.....	37
4 РОЗРОБКА БІЗНЕС-АРХІТЕКТУРИ ІС ДЛЯ АНАЛІЗУ ПОВІТРЯНИХ ОБ'ЄКТІВ.....	41
4.1 Відомості про бізнес модель.....	41

4.2 Метод ВРNM.....	42
4.3 Метод АТАМ.....	46
4.3.1 Стейкхолдери системи.....	46
4.3.2 Обмеження.....	47
4.3.3 Атрибути якості.....	47
ВИСНОВКИ.....	50
ПЕРЕЛІК ПОСИЛАНЬ.....	51
ДОДАТОК А, слайди презентації.....	55
ДОДАТОК Б, сертифікат.....	61
ДОДАТОК В, відомість кваліфікаційної роботи.....	62

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

API - Application Programming Interface, Прикладний програмний інтерфейс

AI - Artificial Intelligence

OCR - Optical character recognition, Оптичне розпізнавання символів

RGB - Red, Green, Blue, адитивна колірна модель

ML, MN - Machine Learning, Машинне навчання

ІС – Інформаційна система

ЛО - літальний об'єкт

ПО – Повітряний об'єкт

ЦАЗ - цифровий аналіз зображення.

ШІ Штучний інтелект

ВСТУП

Машинне навчання та штучний інтелект займають помітне місце в сучасних технологіях. Машинне навчання дозволяє аналізувати великі обсяги даних, на роботу з якими людині знадобилося б дуже багато часу.

Розпізнавання та аналіз об'єктів навколишнього середовища за допомогою цих інструментів в контексті робототехніки – вузький, але ще більш інноваційний напрямок. Прикладом можуть бути роботи Spot від компанії Boston Dynamics, куди вже окрім аналізу навколишнього середовища, вбудували ChatGPT.

В даній же роботі відбувається дослідження більш простих, базових можливостей розпізнавання об'єктів в контексті літальних радіоапаратів.

Також в кваліфікаційній роботі буде представлено розробку технічної та бізнес-архітектури запропонованої інформаційної системи, що передбачає вирішення задачі розпізнавання літальних об'єктів.

1 ОПИС БАЗОВИХ ПОНЯТЬ ДЛЯ АНАЛІЗУ ОБ'ЄКТІВ

1.1 Поняття цифрового аналізу зображень

Аналіз зображень — це отримання інформації із зображень, в основному з цифрових, за допомогою методів обробки цифрових зображень. Завдання аналізу зображення можуть бути простими, як, наприклад, читання тегів зі штрих-кодом, та складними, як, наприклад, ідентифікація людини за обличчям.

Незамінними для аналізу великих обсягів даних, для завдань, які вимагають складних обчислень, для швидкої обробки великої кількості інформації є комп'ютери. З іншого боку, кора головного мозку людини є чудовим апаратом для аналізу зображень, особливо для отримання інформації вищого рівня, та для багатьох застосувань, включаючи медицину, безпеку та дистанційне зондування, людська аналітика все ще не може бути замінена комп'ютерами. З цієї причини багато важливих інструментів аналізу зображень, таких як детектори країв і нейронні мережі, натхненні моделями зорового сприйняття людини.

Цифровий аналіз зображення (ЦАЗ) або аналіз комп'ютерного зображення – це коли комп'ютер (ПК) або електронний пристрій автоматично вивчає зображення, щоб отримати з нього корисну інформацію.

Аналіз комп'ютерного зображення охоплює області комп'ютерного або машинного зору, медичну візуалізацію, і активно використовує розпізнавання образів, цифрову геометрію та обробку сигналів. Ця галузь інформатики була заснована у 1950-х роках у таких академічних установах, як МІТ А.І. лабораторія, спочатку як галузь штучного інтелекту та робототехніки. А в 1950-1970-х роках розвинулась завдяки піонерським внескам Азріеля Розенфельда, Герберта Фрімена, Джека Е. Брезенхема або Кінг-Сун Фу. Тут вже мається на увазі кількісна або якісна характеристика двовимірних (2D) чи тривимірних (3D) цифрових зображень. Наприклад, 2D-зображення

підлягають аналізу в системі комп'ютерного зору, а 3D-зображення – у медичній візуалізації. [23]

Застосування ЦАЗ постійно розширюється в усіх сферах науки та промисловості, включаючи астрономію (обчислення розміру планет), автоматизовану ідентифікацію видів (видів рослин і тварин), медицину, матеріалознавство, робототехніку, військову справу, безпеку, тощо.

В якості прикладу можна розглянути рисунок 1.1, який містить в собі кадр із презентації компанії Майкрософт на тему цифрового аналізу зображення. Концепція полягає у тому, що навченому алгоритму надається фото бобра. На що алгоритм видає в цифрах вірогідність якихось фактів. Тобто, для нього на 100% вірогідно що на зображенні тварина, хордова, ссавець. І з доволі високою вірогідністю, 73% - це бобер. І навіть є уточнення, що це на 71% американський бобер. Такі дані, отримані з “навченої машини” обробляються вже більш простим ПЗ для групування цих даних та подачі їх в більш інтуїтивному, читабельному для людини виді.

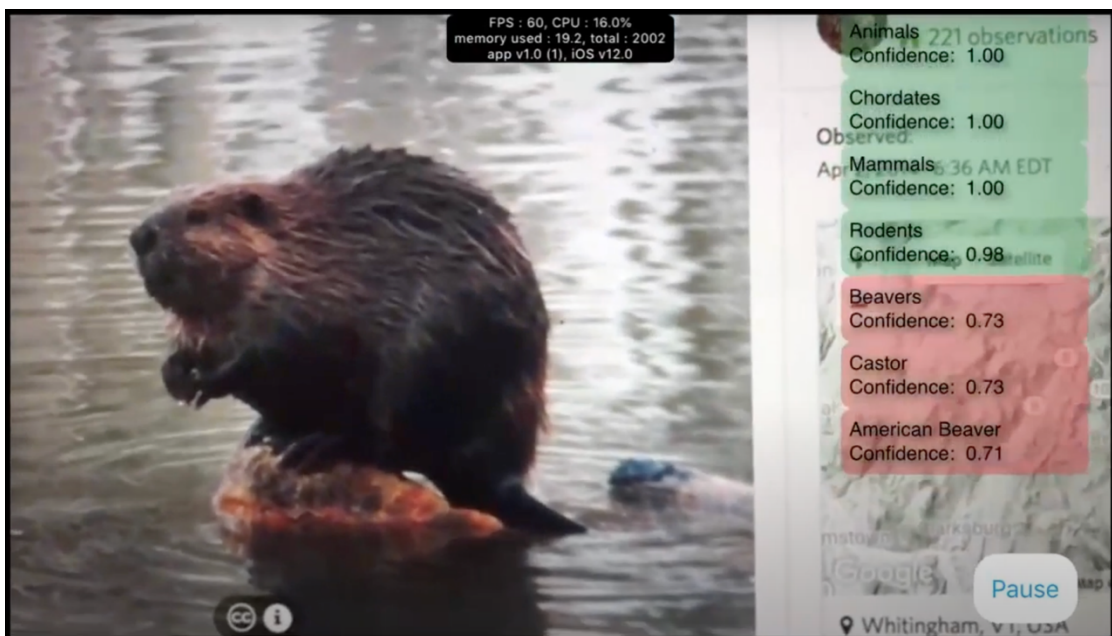


Рисунок 1.1 - Приклад ідентифікації зображення з презентації Microsoft

Ще один приклад - автоматичне розпізнавання номерних знаків, яке використовується при аналізі відео з камер, та синхронізація з базою для,

наприклад, отримання даних водія. (рис.1.2) Або сканування обличчя водія разом з номером. Такі системи зазвичай використовують інфрачервоне освітлення, щоб дозволити камері зробити знімок у будь-який час дня чи ночі.



Рисунок 1.2 - Ідентифікація авто за номером [21]

Насьогодні популярним є оптичне розпізнавання символів або оптичний зчитувач символів (Optical character recognition, OCR) — це електронне або механічне перетворення зображень надрукованого, рукописного або друкованого тексту в машинно-кодований текст зі сканованого документа, фотографії документа, фотографії сцени (наприклад, текст на вивісках чи рекламних щитах у альбомній фотографії) або з тексту субтитрів, накладеного на зображення (наприклад: з телевізійної трансляції). [19]

ЦАЗ також широко використовується як форма введення даних із друкованих паперових записів даних – будь то паспортні документи, рахунки-фактури, банківські виписки, комп'ютеризовані квитанції, візитні картки, пошта, друковані дані чи будь-яка відповідна документація – це поширений метод оцифрування друкованих текстів, щоб їх надалі можна б було електронно редагувати, зберігати більш компактно, відобразити в Інтернеті та використовувати в машинних процесах, таких як когнітивні обчислення,

машинний переклад, (витягнуте) перетворення тексту в мовлення, ключові дані та аналіз тексту.

Ранні версії потрібно було навчати із зображеннями кожного символу та працювати над одним шрифтом за один раз. Удосконалені системи, здатні надавати високу точність для більшості шрифтів, зараз є звичайними, а також підтримують різні вхідні формати файлів зображень.

Деякі системи здатні відтворювати відформатований вихід, який дуже наближений до оригінальної сторінки, включаючи зображення, стовпці та інші нетекстові компоненти.

1.2 Технічна розуміння аналізу зображень

1.2.1 Колірна модель

На найнижчому рівні комп'ютери працюють з 1 і 0. Це не відрізняється при роботі з зображеннями. ПК можуть уявляти зображення у вигляді матриці одиниць і нулів. Коли ПК відображає зображення, він відображає сітку пікселів, кожен з яких заповнений одним кольором. Наприклад (рисунок 1.3), дуже проста форма, така як чорний трикутник нижче. [22]



Рисунок 1.3 - Простий трикутник із низькою роздільною здатністю, що складається з 28 пікселів

Ця форма складається з 28 пікселів: 16 чорних пікселів і 12 білих пікселів, кожен з яких можна побачити, якщо представити зображення як

показано на рисунку 1.4.

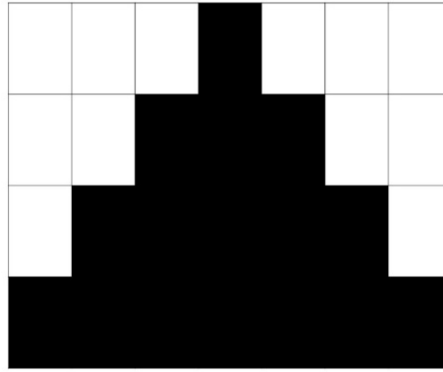


Рисунок 1.4 - Чорний трикутник із контуром кожного пікселя

Але це зображення є лише представленням матриці чисел, що зберігається комп'ютером (рисунок 1.5).

0	0	0	1	0	0	0
0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1

Рисунок 1.5 - Демонстрація 1 і 0 у матриці, яка утворює зображення трикутника

Це досить просто для зображень лише з двома кольорами: чорним і білим. Якщо образ у відтінках сірого, потрібно додати більше нюансів. Натомість у полях можуть опинитися такі числа, як 0,015. Це число, по суті, повідомляє комп'ютеру, яку яскравість надавати кожному пікселю на екрані. Він визначає значення між чорним (1) і білим (0).

Якщо трикутник стане зеленим, аналіз стане більш детальним за 1 та 0. У цьому випадку не вийде використовувати число від нуля до одиниці для опису цього кольору. Доступних кольорів надто багато, щоб практично кожному з них призначити окреме число, яке було б інтуїтивно зрозумілим. Замість цього було розроблено кілька систем, які дозволяють розділити окремі

компоненти кольору та призначити номери кожному компоненту.

Наприклад, колірна модель RGB, яка розбиває кожен колір на три основних кольори (червоний, зелений і синій), які поєднуються, щоб створити потрібний колір. Вона дозволяє використовувати тип додаткового змішування кольорів, який використовується світлом, у телевізійних пристроях.

У цій системі основні кольори називаються «каналами». Для 8-бітного зображення кожному каналу присвоюється значення від 0 до 255 (це максимальне значення, яке може мати 8-значне двійкове число).

Значення RGB для відтінку зеленого становлять 154, 205 і 50 відповідно. Якби це був чистий зелений колір, червоний і синій канали мали б значення 0, а зелений - 255. Таке представлення RGB показує, що для створення цього відтінку в зелений змішано багато червоного та трохи синього. Кожен піксель має три канали значень, які його описують. Кожен зелений піксель позначається як 154, 205, 50, а кожен білий піксель – як 0, 0, 0. Це означає, що цей простий 28-піксельний зелений трикутник (рисунок 1.6) буде представлений трьома матрицями 7x4 (рисунок 1.7). [22]

Крім RGB існує ще CYMK система, яка використовує чотири числа для опису кожного кольору та чотири кольори: блакитний, пурпурний, жовтий і чорний. Це корисно в ситуаціях, коли потрібна субтрактивна колірна модель, наприклад, під час друку.



Рисунок 1.6 - 28-піксельний трикутник зеленого кольору

Red Channel							Green Channel							Blue Channel						
0	0	0	154	0	0	0	0	0	0	205	0	0	0	0	0	0	50	0	0	0
0	0	154	154	154	0	0	0	0	205	205	205	0	0	0	0	50	50	50	0	0
0	154	154	154	154	154	0	0	205	205	205	205	205	0	0	50	50	50	50	50	0
154	154	154	154	154	154	154	205	205	205	205	205	205	205	50	50	50	50	50	50	50

3 Matrix Stack

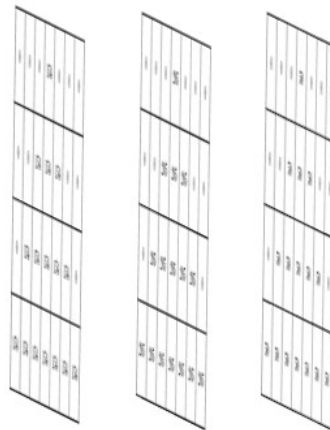


Рисунок 1.7 - Три матриці при поєднанні утворюють одне зображення

1.2.2 Ізоляція об'єктів за допомогою порогів

Для того, щоб розпізнати зображення з відео, відео розглядається як серія кадрів, складених разом. І якщо є одне фото в градаціях сірого, то необхідно вказати ПК, яка частина фотографії є ящіркою, а яка ні (рисунок 1.8).

Найпростішим способом є відмітити пікселі вручну, клікаючи на кожний мишею по черзі. Але враховуючи роздільну здатність більшості фотографій і кількість фотографій, які входять до одного відео, ця задача буде досить трудомісткою.

Тому є автоматизований спосіб – це задатися правилом, за яким можна визначати об'єкти.



Рисунок 1.8 - Двійкове зображення, створене шляхом встановлення порогового значення 130

Найпростіший спосіб є використання порогів.

Порогове визначення – це процес знаходження краю об’єкта, в даному випадку ящірок, використовуючи різницю в яскравості. Ця техніка спрямована на визначення порогової яскравості, яка розділяє світлі та темні пікселі.

Уявімо матрицю чисел, яка визначає це зображення у відтінках сірого. Кожен піксель визначається числом від 0 до 255. За допомогою порогів можна визначити правило, що будь-що досить темне є у даному випадку ящіркою. Наприклад, можна сказати, що будь-який піксель зі значенням 130 або вище є ящіркою, а будь-який піксель нижче 130 не є ящіркою (рисунок 1.9).

Технічно може бути взято будь-що з обраними пороговим значенням або вище на 255, а все інше на 0. Це створює бінарне зображення з чорними та білими пікселями. Якщо вибране правильне порогове значення, ящірки будуть чорними на білому фоні.

У цьому прикладі як порогове значення обрано довільне число 130. Однак, щоб точно ізолювати ящірок, потрібно обрати число, яке охопило б переважну більшість ящірок, мінімізуючи пікселі, які не є її частиною. Цей процес вибору одного порогового значення для всього зображення називається глобальним пороговим значенням.

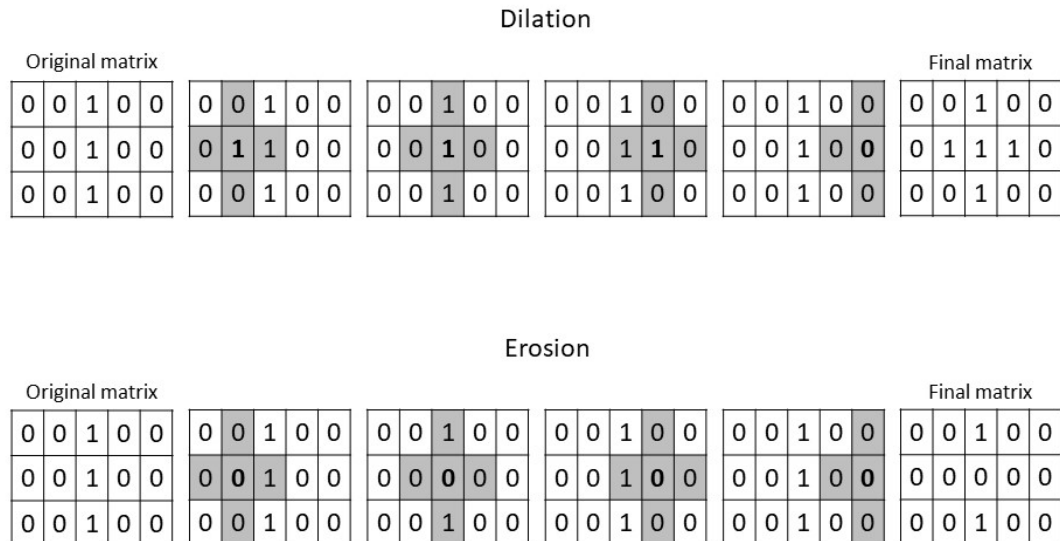


Рисунок 1.9 - Візуальний приклад розширення та ерозії за допомогою структурного елемента у формі хреста

На рисунку виконано морфологічну операцію лише над другим рядком матриці. У реальному зображенні його буде застосовано до кожного пікселя.

Даний спосіб є недосконалим. Якщо в наявності об'єкт з ідеальним силуетом на ідеально білому фоні, глобальне порогове визначення буде працювати добре. Але реальні зображення рідко бувають такими простими, а фон ідеальним.

Тому використовують більш складний метод визначення порогу - локальний поріг. У цій техніці обираються різні порогові значення для різних частин зображення. Це особливо корисно, коли є градієнт освітлення на фоні.

Існує багато інших методів оптимізації, які відрізняються за складністю, і той, який буде обраний, залежатиме від обставин. [22]

1.2.3 Розпізнавання об'єкту за допомогою морфологічних операцій

За допомогою порогового значення визначається правило на основі кольору чи освітленості. Іншим способом є визначення правила на основі форми за допомогою морфологічних операцій.

Морфологічні операції використовують елементи структури для

визначення країв фігур. Структурний елемент — це невелика матриця певної форми, як правило, квадрат, коло або хрест (рисунок 1.10). Щоб отримати різні ефекти можна змінити розмір структурного елемента.

Під час морфологічної операції кожен піксель порівнюється зі своїми сусідами в цій формі. Є дві основні морфологічні операції, які можна використати для визначення форми наміченого об'єкта: розширення та ерозія.

Ерозія діє так само, як розширення, але з протилежним ефектом. З ерозією, якщо будь-який із сусідів пікселя є 0, тоді цей піксель буде перевизначено як 0. Це може служити для розділення об'єктів, зменшення шуму та уточнення меж. [22]

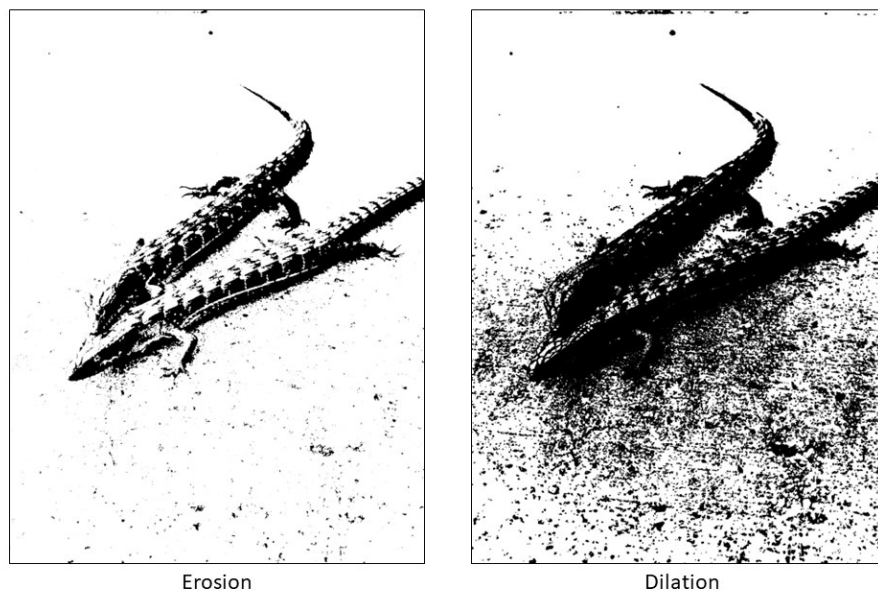


Рисунок 1.10 - Ерозія та розширення, застосовані до фотографії ящірки (структуруючим елементом використано квадрат 6х6)

На рисунку показано, як розширення та ерозія змінюють бінарне зображення ящірки.

Розширення та ерозію можна використовувати ітеративно або послідовно для зміни кінцевого продукту. Звичною практикою є використання як розширення, так і ерозії для подальшого визначення ізольованого зображення. (рисунок 1.11)

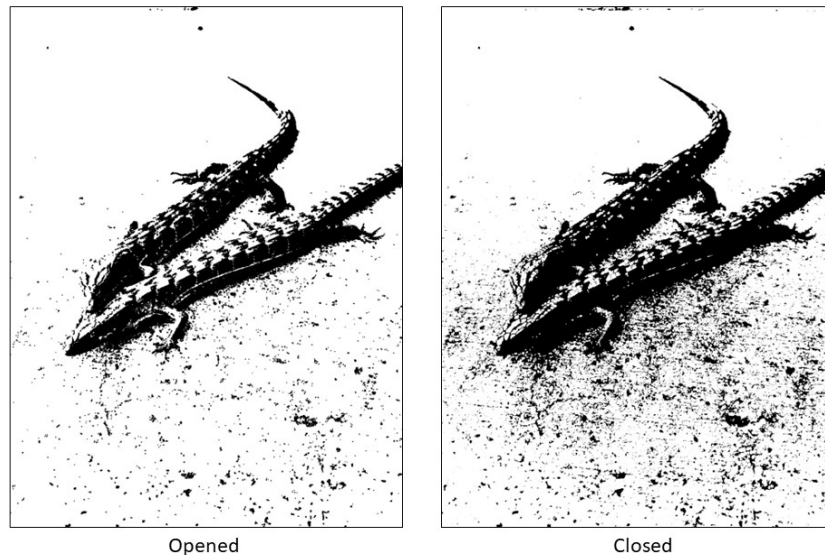


Рисунок 1.11 - Демонстрація техніки відкриття та закриття з використанням квадратного структурного елемента 6x6 на фотографії ящірки

Коли ерозія супроводжується розширенням, невеликі виступи і шуми видаляють, а контур, що залишився, зміцнюють. Ця техніка називається відкриттям.

Коли розширення супроводжується ерозією, невеликі дірки та щілини заповнюються, а краї підводяться в порядок. Цей прийом називається закриттям.

Можна побачити, що жодне з цих зображень повністю не ізолює ящірок від фону. Але з кожним налаштуванням і повторенням можна ставати ближче й ближче.

З наступною ітерацією можна наказати комп'ютеру ігнорувати все, що виходить за межі певного діапазону розмірів. Це видалить більшість крапок і залишить об'єкти недоторканими.

Для кожного зображення або серії зображень можна налаштувати серію таких правил, щоб повідомити ПК, як повинна виглядати ціль. Тоді комп'ютер може використовувати цю інформацію для прийняття рішень.

Ці рішення впливають на все, від ідентифікації галактик до пошуку хворих тканин. Але створення серії правил для кожного окремого зображення займає багато часу та втомлює. Для вирішення такої задачі і використовується машинне навчання. [22]

1.3 Класифікація зображень

Класифікація зображень - це фундаментальне завдання, метою якого є присвоєння мітки або класу вхідному зображенню. Для класифікації, потрібно мати модель, за якою вона буде відбуватись. Під час навчання, модель ітеративно коригує свої параметри, використовуючи методи оптимізації, наприклад, градієнтний спуск, щоб мінімізувати різницю між її прогнозами та базовими мітками істинності, які визначились раніше. Цей процес включає обчислення функції втрат, яка кількісно визначає помилку передбачення моделі, і відповідне оновлення параметрів.

Навчання керованої моделі (ML) для аналізу зображень є ітеративним процесом. Модель вчиться за допомогою позначеного набору даних, оцінюються її продуктивність у наборі перевірки та вносяться коригування для підвищення її точності. Цей ітераційний цикл триває, поки модель не досягне задовільних результатів. Це є контрольована ML (supervised ML), або контрольоване навчання.

Після того, як контрольовану модель ML було навчено на позначених даних зображення, її можна застосовувати до різних завдань аналізу зображень. Нижче наведено деякі поширені програми, які підкреслюють ефективність керованих моделей машинного навчання в задачах обробки зображень.

Наприклад, контрольовану модель ML можна навчити класифікувати зображення та розрізнити наприклад їжу та столові прилади на фотографії яєшні. Використовуючи різноманітний набір даних із позначеними прикладами, модель може узагальнювати та точно класифікувати зображення з високим рівнем точності.

Цю техніку можна використовувати для таких завдань, як визначення хвороб на медичних зображеннях, розпізнавання об'єктів на супутникових знімках і навіть класифікація емоцій за виразом обличчя, тощо.

Одним з важливих застосувань моделей ML в аналізі зображень є

виявлення об'єктів. На відміну від класифікації зображень, виявлення об'єктів включає не лише ідентифікацію об'єктів, присутніх на зображенні, але й визначення їх місцезнаходження шляхом малювання обмежувальних рамок навколо них.

Цей процес дозволяє моделі ідентифікувати та вкладено класифікувати дрібні об'єкти на зображенні. Наприклад, на рисунку 1.12, як і було обговорено раніше, на фото з яєшнею ідентифікується їжа та столові прилади. [22].

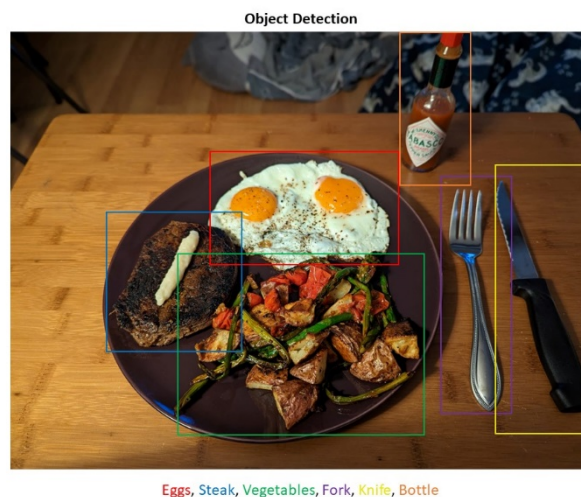


Рисунок 1.12 - Приклад виявлення об'єкта

Окрім описаної неконтрольованої ML існує ще неконтрольована ML (unsupervised ML). Неконтрольоване навчання не покладається на позначені дані, а спрямоване на виявлення прихованих закономірностей, структур або зв'язків у самих даних.

Мета неконтрольованого навчання при аналізі зображень полягає в тому, щоб виявити значущі структури та ідеї з немаркованих даних зображення. Використовуючи такі методи навчання, можна отримати цінну інформацію та отримати глибше розуміння основних характеристик зображень.

Неконтрольоване навчання може допомогти ідентифікувати кластери схожих зображень, виявити візерунки чи текстури, характерні для певних класів зображень, і виявити аномалії чи викиди в даних. [22]

2 ОГЛЯД ТА АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

2.1 Використання машинного навчання у військовій справі

Машинне навчання (МН) широко використовується в різних галузях, в тому числі і у військовій. Системи, що використовують технології машинного навчання та штучного інтелекту (ШІ), у порівнянні з системами, що їх не використовують, здатні ефективніше обробляти великі обсяги даних. Крім того, ШІ покращує самоконтроль, саморегуляцію та самозапуск систем завдяки властивим йому обчислювальним можливостям і можливостям прийняття рішень.

ШІ та МН розгортається майже у всіх військових програмах [17], а збільшення фінансування досліджень і розробок військовими дослідницькими агенціями обіцяє ще більше сприяти застосуванню систем на основі ШІ у військовому секторі.

Наприклад [19], Агентство передових оборонних дослідницьких проєктів (Defense Advanced Research Projects Agency, DARPA) Міністерства оборони США фінансує розробку роботизованої системи підводного човна, яка, як очікується, використовуватиметься в різноманітних програмах від виявлення підводних мін до участі в водних операціях. Крім того, Міністерство оборони США витратило 7,4 мільярда доларів на штучний інтелект, великі дані та хмарні технології в 2017 фінансовому році. Очікується, що до 2025 року розмір ринку військових рішень для машинного навчання досягне 19 мільярдів.

Крім того, корисним є перекладання тексту в реальному часі. Тут комп'ютери можуть навчитись виділяти важливі поняття та частини тексту з великих баз даних тексту, зібраного з різних медіаджерел, після цього використовувати ці “знання” для подальших перекладів. Тому одним з нещодавно зареєстрованих проривом у ML є точний переклад тексту між різними мовами.

Ще однією ваговою областю для машинного навчання є виявлення аномалій. Традиційно, це виконується шляхом спочатку ідентифікації кластерів відомих даних і характеристики розподілу, під яку потрапляють дані. Потім, у міру обробки нових вхідних даних, вони або ідентифікуються як такі, що потрапляють до, або поза вихідними розподілами. Якщо вони знаходяться за межами відомих розподілів, вони вважаються аномаліями. Багато з наступних типів систем виявлення аномалій можуть бути корисні й у військовій справі:

- виявлення кіберввторгнення: мережевий трафік, який виходить за межі звичайного. McPAD і PAYL77 є двома такими прикладами програмного забезпечення, яке зараз використовується, та яке використовує виявлення аномалій;
- патерни життєвих аномалій: візуальні та біометричні зображення людей, які діють не звичайним чином, що свідчить про те, що вони можуть виконувати певні дії;
- технічне обслуговування за умовами: сигнали, які не є типовими для матеріалу/системи в його віці в поточному життєвому циклі;
- аномалії людини: підстави вважати, що біометрія людини вийшла з ладу;
- виявлення сторонніх предметів: візуальні зображення об'єктів, не розпізнаних у базі даних відомого матеріального забезпечення.

Актуальним є також питання робототехніки. Використання машинного навчання в цій галузі також є дуже обширною сферою. Сфери, де машинне навчання продовжує мати сенс, включають зондування, навігацію, пересування та прийняття рішень.

Окрім робототехніки, бажано, щоб будь-яка система самостійно поверталася до нормального робочого стану, коли пошкоджена або не працює на повну потужність. Для цього потрібен інтелект певного рівня, щоб автономно діагностувати недоліки та проблеми та виправляти ці проблеми за допомогою наявних ресурсів. [18]

2.2 Використання аналізу зображень у військовій справі

У військовій справі важливою та актуальною задачею є захоплення та супроводження цілей, їх автоматичне розпізнавання, ідентифікація та ведення.

Програми обробки зображень військового характеру включають сцени відкритого світу у необмеженому середовищі. Ці проблеми призводять до необхідності розробки та використання складних і витончених алгоритмічних рішень.

Обмеження реального часу багатьох додатків висувають дуже високі обчислювальні вимоги до системи обробки зображень. Файли журналу з обох систем відстеження передаються через канал передачі даних до системи дистанційного аналізу в режимі реального часу або як пакетна передача після завершення тесту. Система визначення відстані також може працювати в режимі реального часу без постаналізу.

Як було визначено раніше, обробка зображення — це метод виконання деяких операцій із зображенням, щоб отримати покращене зображення або витягти з нього деяку корисну інформацію. Тобто, це тип обробки сигналу, у якому входом є зображення, а виходом може бути зображення або характеристики/функції, пов'язані з цим зображенням.

На даний момент є наступні перспективні актуальні напрямки використання технології у військовій справі:

- виявлення та відстеження рухомих та нерухомих об'єктів;
- наведення ракети або дрона-камікадзе на ціль;
- навігація автомобіля;
- широкомасштабне спостереження;
- система розуміння зображень;
- обчислювальний зір;
- дистанційне зондування оточення. [18].

2.3 Огляд сервісів, що надають послуги ML/AI

Під час аналізу ринку, було виявлено наступні проекти, ціль та послуги яких перетинаються з поточною роботою:

- griselda.com.ua – як заявлено на вебсайті, “Автоматизована система внесення, обробки та передачі інформації з використанням штучного інтелекту”, Гріселда – це молодий перспективний сервіс з надання послуг обробки даних за допомогою ШІ. Прикладом використання системи є мапа бойових дій в Україні – Deepstate map. Вони використовують ШІ аналітику для аналітики переміщення військ супротивника. [31]

- saker.airforce – ІС, що побудована на алгоритмах штучного інтелекту. Як заявлено у відкритих джерелах, система за допомогою спеціальної оптики самостійно розпізнає і фіксує координати техніки ворога, навіть у випадку маскування, передаючи інформацію в командний пункт прийняття рішень. Також, дрон може використовувати інерціальну систему наведення, що значно підвищує його стійкість до засобів РЕБ. [32]

- shield.ai – Компанія, що займається розробкою пілотів та ко-пілотів на базі штучного інтелекту, надаючи рішення як цивільним так і військовим. “Відсутність талантів у розробці програмного забезпечення та штучного інтелекту у великих оборонних виробників дала можливість почати все заново та побудувати компанію 21-го сторіччя в сфері аерокосмічних і оборонних технологій, зосереджену на пілотах штучного інтелекту. А в 2015 році разом із співзасновниками Райаном Ценгом та Ендрю Райтером було засновано Shield AI, щоб надати Міністерству оборони та союзникам найкращі технології штучного інтелекту та автономності для захисту військовослужбовців і цивільних.”, - кажуть самі про себе shield.ai [33]

- chat.openai.com – вважається, що найкращий референс навіть для рішення на базі ChatGPT – це OpenAI, розробники ChatGPT, тому що вони теж так чи інакше надають послугу у вигляді доступу до обчислень на їх навченому ШІ. Більш того, вони є прикладом зручної гнучкої монетизації та

простою системою доступу до їх API, що теж корисно при розробці свого рішення. [34]

- neptune.ai – приклад більш загальної B2B моделі щодо оренди інструментарію по роботі з ШІ та МН. Надає можливість для реалізації експериментів для команд машинного навчання, ділитися результатами та розбирати питання по мірі виникнення. Сервіс пропонує єдине місце для відстеження, порівняння, зберігання та спільної роботи над експериментами та моделями. [35]

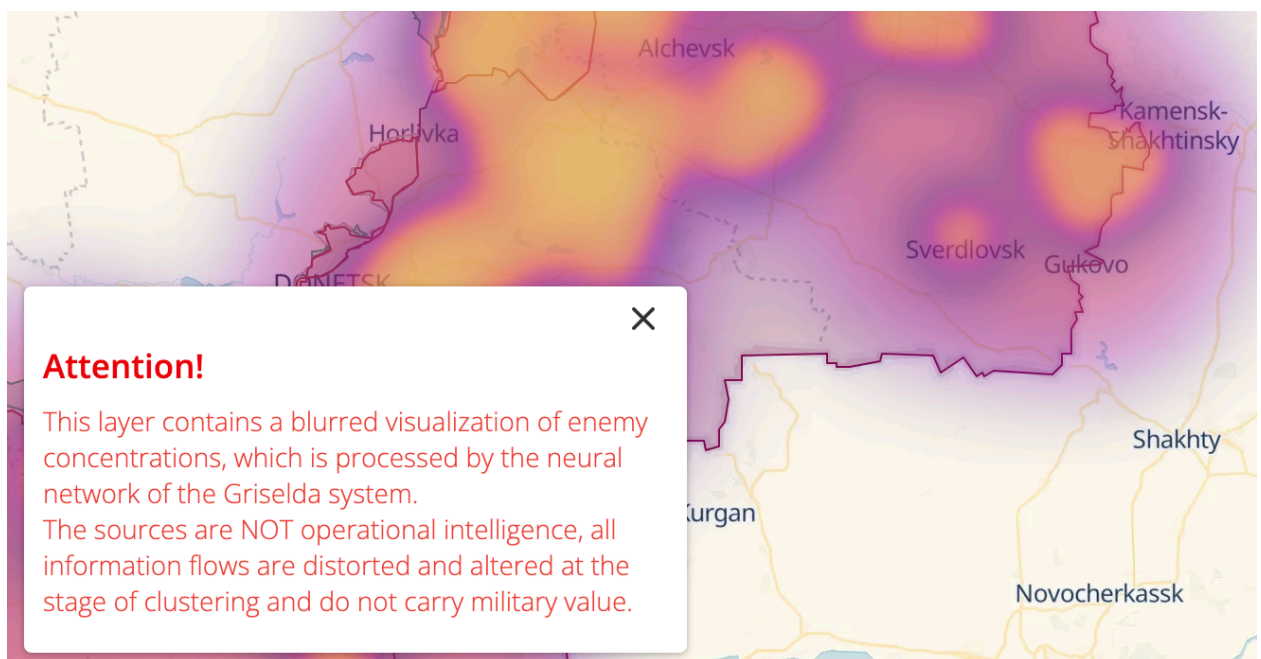


Рисунок 2.1 – Приклад використання Griselda в сервісі Deepstate Map [8]

На підставі огляду вище, можна сказати що тема ШІ та машинного навчання на даний момент розвивається доволі активно та доволі затребувана в поточних реаліях. Тому дана робота присвячена розробці архітектури системи для аналізу повітряних об'єктів з використанням можливостей машинного навчання.

За конкурентну здібність буде взята гнучкість та підтримуваність системи, можливість налаштовувати під різні набори даних, або паралельно навчити систему виявляти та класифікувати об'єкти різних типів.

3 РОЗРОБКА ТЕХНІЧНОЇ АРХІТЕКТУРИ ІС ТА ДОСЛІДЖЕННЯ ІНСТРУМЕНТАРІЇВ ДЛЯ ЇЇ РЕАЛІЗАЦІЇ

3.1 Розробка функціональних вимог до проєктуємої інформаційної системи

Функціональні вимоги — це вимоги до системи, які описують внутрішню її роботу та поведінку: обчислення даних, маніпулювання даними, обробка даних та інші специфічні функції, які має виконувати система. На відміну від нефункціональних вимог, які визначають якою система повинна бути, функціональні вимоги визначають, що система повинна робити. Функціональні вимоги до програмного забезпечення визначаються на першій стадії процесу розробки ІС — на етапі аналізу вимог.

На рисунку 3.1 розглянуті функціональні вимоги до інформаційної системи аналізу повітряних об'єктів.

ІС має три сутності: літальний об'єкт (ЛО), програмне забезпечення та машинне навчання.

ЛО повинно забезпечувати можливість фізичного стеження за об'єктом та виконання команд оператора.

ПЗ повинно реалізовувати можливості виконання команд на наведення на об'єкт, відстеження об'єкту передачі кадрів зображення на подальший аналіз, обробку отриманої інформації та інформування оператора.

Можливості ML повинні забезпечувати виконання аналізу отриманих кадрів та надавати результати у цифровій формі.

На рисунку 3.2 представлені пояснення умовних позначень до рисунку 3.1.

Функціональні вимоги

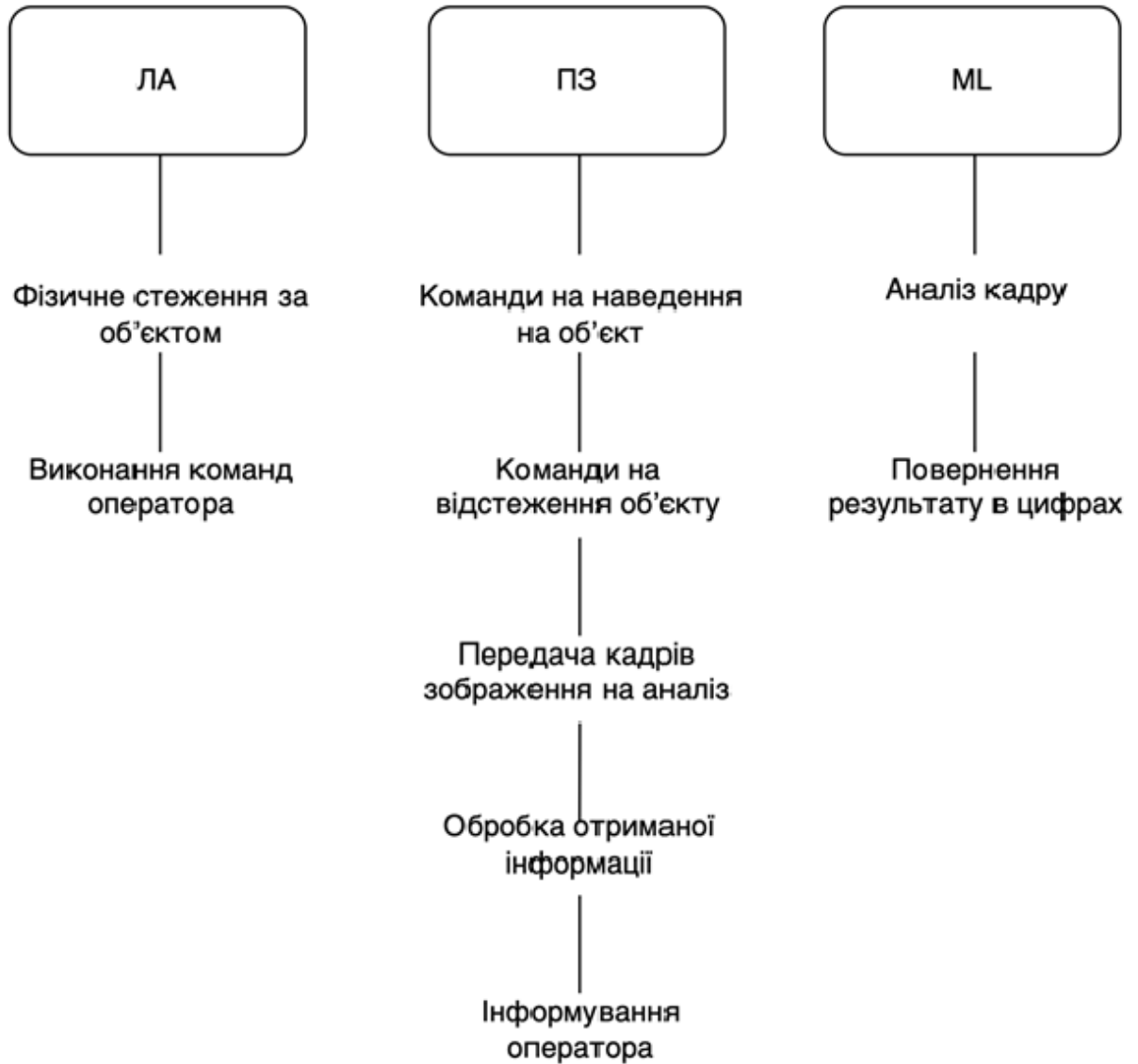


Рисунок 3.1 – Функціональні вимоги до інформаційної системи аналізу повітряних об'єктів

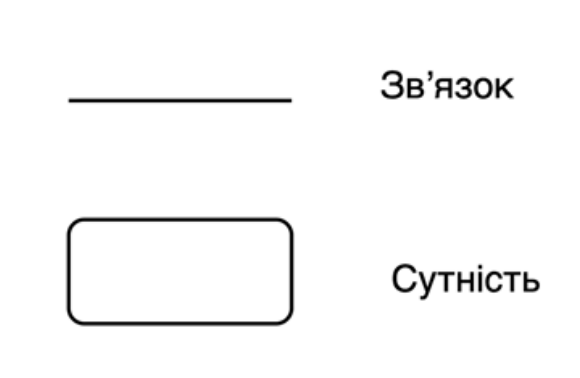


Рисунок 3.2 – Перелік умовних позначень до рисунку 3.1

3.2 Архітектурний концепт ІС для аналізу повітряних об'єктів

Взаємодія між основними компонентами ІС, показана на рисунку 3.3. А пояснення умовних позначень до неї - на рисунку 3.4.

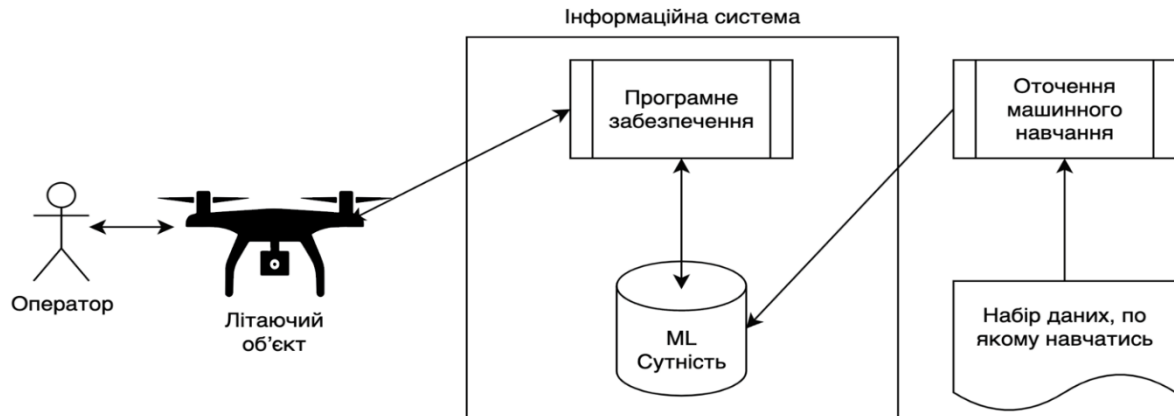


Рисунок 3.3 – Взаємодія між елементами ІС для аналізу повітряних об'єктів



Рисунок 3.4 – Перелік умовних позначень до рисунку 3.3

Фізична взаємодія налічує 4 учасники процесу:

- ноутбук вивантажує ПЗ з ІС через USB на літальний об'єкт;
- літальний об'єкт виводить інформацію, в тому числі з ІС, на екран або окуляри;
- оператор, отримуючи інформацію з екрану або окуляра, використовує контролер як засіб введення.

Схематично цей процес зображений на рисунку 3.5.

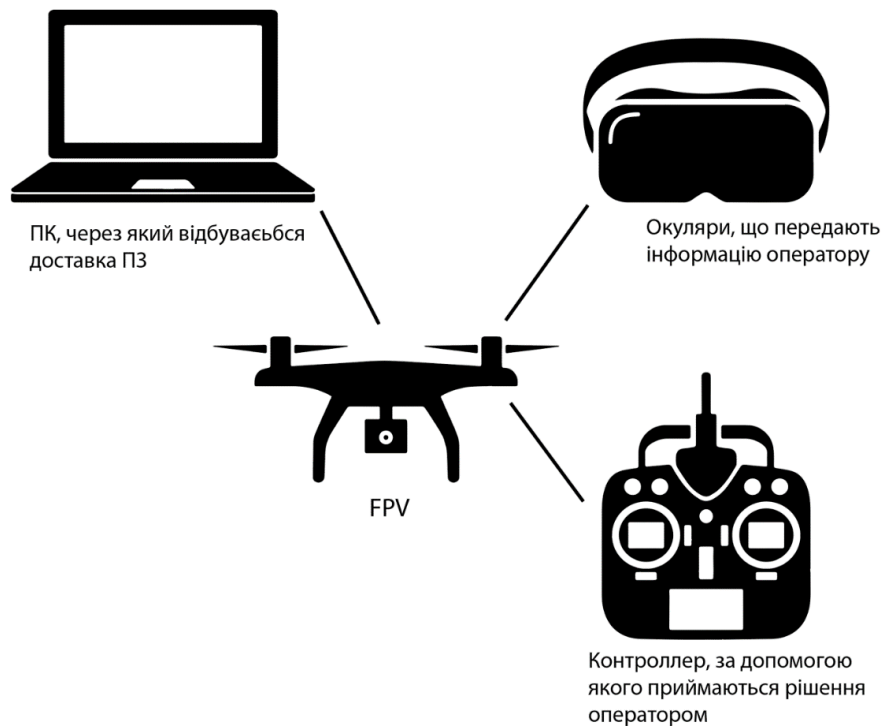


Рисунок 3.5 – Фізична взаємодія об'єктів у ІС

3.3 Огляд можливостей ChatGPT для реалізації задачі аналізу повітряних об'єктів

Для виконання задачі проектування інформаційної системи аналізу повітряних об'єктів з використанням машинного навчання спочатку необхідно обрати рушій, який буде виконувати.

Початково, розглядалися основні рушії, які будуть виконувати аналітичну функцію - це ШІ рушій ChatGPT та сутності машинного навчання.

Спершу для тестування через популярність та активний розвиток і відносно легку монетизацію, був обраний ChatGPT. ChatGPT - це велика статистична модель мови, оптимізована для ведення діалогів та відлагоджена завдяки технікам керованого навчання та навчання з підкріпленням. В основі прототипу лежить модель OpenAI GPT-3.5 — покращена версія GPT-3.

Прототип ChatGPT було випущено 30 листопада 2022 року. Через детальність та ясність відповідей, його популярність виросла неймовірно швидко, хоча фактична точність цих відповідей підлягала критиці.

Після релізу ChatGPT оцінка компанії OpenAI виросла до 29 мільярдів доларів. Лише за два місяці після випуску, кількість активних користувачів перевищила 100 мільйонів — такий короткий проміжок став історичним рекордом серед користувацьких програм.

Під час тестування були виконані намагання "навчити" ChatGPT робити глибоку аналітику в цифрах, прописавши певну риторичну запит із ключових слів та шаблонів динамічних запитів.

Задля тестування концепту, щоб зрозуміти, наскільки рішення підходить до задачі та наскільки просте в імplementації, для ChatGPT було подане просте питання щодо обчислення кількості проданої їжі. Була також умова щодо формату виводу, щоб маючи налаштоване API, мати можливість легко обробляти відповіді від ChatGPT та передавати далі, наприклад у фронтенд, або кешувати за потребою.

Для розуміння доступних можливостей, які надасть сервіс від OpenAI, було прийняте рішення створити відразу акаунт розробника.

Потім був налаштований зв'язок з ChatGPT API в застосунку Postman — популярної утиліти для HTTP запитів.

На рисунку 3.6 зображено фрагмент з програми Postman — ПЗ для формування HTTP запитів, де зображений запит на підрахунок простої інформації, пов'язаної з прогнозуванням потенційних координат, наприклад.

За результатами можна побачити, що ChatGPT справився із задачею задовільно, зберігши формат, який потім можна опрацювати за допомогою програмного коду та витягнути потрібну інформацію простою обробкою строки.

Але в подальшому просуванні та виставленні більш складних риторик та задач, ШІ від OpenAI міг губитись або з оформленням відповіді, або з фокусом на чомусь конкретному, упускаючи додаткові фактори, важливі для загальної картини.

Також стало питання “конвертації” даних які потрібно буде аналізувати, коли масштаби цих даних будуть великі. Іншими словами, як передати ChatGPT на аналіз набір даних із декількох таблиць на 50 Мегабайт.

```

"role": "user",
"content": "Please help me to solve this puzzle. In the previous year the customers of our cafe bought burgers with the next
rate month 1 - 1234, month 2 - 1253, month 3 - 1299, month 4 - 1211, month 5 - 1212, month 6 - 1218, month 7 - 1222, month 8
- 1229, month 9 - 1235, month 10 - 1240, month 11 - 1222, month 12 - 1221. Please estimate the next year for the burger
selling"
}

Headers (23) Test Results Status: 200 OK Time: 17.81 s Size: 2.86 KB Save as Example

Preview Visualize JSON

"content": "To estimate the burger selling for the next year, we can analyze the trend in the previous year's data.
\n\nFirst, let's calculate the average number of burgers sold per month in the previous year:\n\n(1234 + 1253 + 1299
+ 1211 + 1212 + 1218 + 1222 + 1229 + 1235 + 1240 + 1222 + 1221) / 12 = 1227.5\n\nThe average number of burgers sold
per month is approximately 1227.5.\n\nNext, let's calculate the average monthly change in burger sales:\n\n(1253 -
1234) + (1299 - 1253) + (1211 - 1299) + (1212 - 1211) + (1218 - 1212) + (1222 - 1218) + (1229 - 1222) + (1235 - 1229)
+ (1240 - 1235) + (1222 - 1240) + (1221 - 1222) / 11 = 1.1818\n\nThe average monthly change in burger sales is
approximately 1.1818.\n\nNow, let's use this average monthly change to estimate the burger selling for the next
year:\n\nMonth 1: 1227.5 + 1.1818 = 1228.6818 (approximately 1229)\nMonth 2: 1229 + 1.1818 = 1230.1818 (approximately
1230)\nMonth 3: 1230 + 1.1818 = 1231.1818 (approximately 1231)\nMonth 4: 1231 + 1.1818 = 1232.1818 (approximately
1232)\nMonth 5: 1232 + 1.1818 = 1233.1818 (approximately 1233)\nMonth 6: 1233 + 1.1818 = 1234.1818 (approximately
1234)\nMonth 7: 1234 + 1.1818 = 1235.1818 (approximately 1235)\nMonth 8: 1235 + 1.1818 = 1236.1818 (approximately
1236)\nMonth 9: 1236 + 1.1818 = 1237.1818 (approximately 1237)\nMonth 10: 1237 + 1.1818 = 1238.1818 (approximately
1238)\nMonth 11: 1238 + 1.1818 = 1239.1818 (approximately 1239)\nMonth 12: 1239 + 1.1818 = 1240.1818 (approximately
1240)\n\nTherefore, the estimated burger selling for the next year is as follows:\n\nMonth 1: 1229\nMonth 2:
1230\nMonth 3: 1231\nMonth 4: 1232\nMonth 5: 1233\nMonth 6: 1234\nMonth 7: 1235\nMonth 8: 1236\nMonth 9: 1237\nMonth
10: 1238\nMonth 11: 1239\nMonth 12: 1240"
}

```

Рисунок 3.6 – Результати відповіді від ChatGPT використовуючи ChatGPT API

В додаток, активний розвиток має і негативні сторони: поточна версія чату вже через пів року застаріє, а ще трохи пізніше перестане підтримуватись. Або зміниться модель розповсюдження/монетизація, що призведе до неочікуваних зайвих витрат фінансів та часу.

Тому було розглянуто альтернативний шлях через використання бібліотек моделей навчання.

3.4 Огляд бібліотек моделей навчання для вирішення задачі аналізу повітряних об'єктів

Для тестування концепту аналізу цілей, потрібно в першу чергу знайти бібліотеку машинного навчання для подальшого використання, навчання.

Було оглянуто наступні популярні рішення [5]:

- SciPy - популярна бібліотекою серед ентузіастів машинного навчання, містить різні модулі для оптимізації, лінійної алгебри, інтеграції та статистики. Існує різниця між бібліотекою SciPy і стеком SciPy. SciPy є одним

із основних пакетів, які складають стек SciPy. SciPy також дуже корисний для обробки зображень.

- Scikit-learn — популярна бібліотека ML для класичних алгоритмів ML. Створена на основі двох основних бібліотек Python, а саме NumPy і SciPy. Scikit-learn підтримує більшість контрольованих і неконтрольованих алгоритмів навчання. Scikit-learn також можна використовувати для інтелектуального аналізу даних, що робить бібліотеку чудовим інструментом для тих, хто починає працювати з машинним навчанням.

- PyTorch — це бібліотека машинного навчання з відкритим вихідним кодом для Python на основі Torch, бібліотеки машинного навчання з відкритим кодом, реалізованої на C з оболонкою в Lua. Тут наданий великий вибір інструментів і бібліотек, які підтримують комп'ютерне бачення, обробку природної мови (NLP) і багато інших програм ML. Це дозволяє розробникам виконувати обчислення на тензорах із прискоренням GPU, а також допомагає створювати обчислювальні графіки.

- Keras — бібліотека машинного навчання для Python. Це високорівневий API нейронних мереж, здатний працювати поверх TensorFlow, CNTK або Theano. Він може безперебійно працювати як на ЦП, так і на ГП. Keras легко дозволяє початківцям ML створювати та проектувати нейронні мережі. Однією з найкращих переваг Keras є те, що він дозволяє легко та швидко створювати прототипи.

- TensorFlow — це популярна бібліотека з відкритим кодом для високопродуктивних чисельних обчислень, розроблена командою Google Brain у Google. Як випливає з назви, Tensorflow — це фреймворк, який передбачає визначення та виконання обчислень за допомогою тензорів. Він може тренувати та запускати глибокі нейронні мережі, які можна використовувати для розробки кількох додатків III. TensorFlow широко використовується в галузі досліджень і застосування глибокого навчання.

Після огляду рішень, була обрана бібліотека Tensorflow. Обумовлено це популярністю бібліотеки, оновлюваністю, кількістю статей та документації,

має відкритий код, безоплатня, та при тому реалізує глибоке навчання.

3.5 Вибір набору даних для реалізації задачі аналізу повітряних об'єктів

Також з урахуванням неоптимальності тестування на простих даних, яке відбулось під час тестування ChatGPT, для розуміння, чи підходить рішення, був здійснений пошук набору даних.

До вимог до набору даних, що підходять для реалізації задачі аналізу повітряних об'єктів, можна віднести наступне:

- велика кількість фотографій – дані повинні складатись з більш ніж 1000 фото;
- велике різноманіття об'єктів – 1000 фото не повинно бути умовно тільки про 2 літака;
- актуальність об'єктів – дуже бажано мати дані про сучасні літаки, хоча б з 2010 року;
- якість фотографій – система повинна навчатись на якісному зображенні;
- опційно, додаткові таблиці з інформацією, наприклад швидкості польоту;
- приклади використання – набір даних не повинен бути «поламаним», для цього бажано мати докази, що на ньому можна «навчитись».

Були розглянуті різні набори даних на різних ресурсах [36], [37], [38].

Фінальним результатом був обраний набір даних був знайдений на ресурсі Kaggle [6].

Це актуальний набір даних по сучасним військовим літакам, що містить в собі великий набір фотографій для кожного літака, а також набір їх характеристик у цифрах, має різні демонстрації використання на ресурсі Kaggle. Загальний розмір набору даних склав 12 Гігабайт.

Фрагменти набору даних представлені на рисунках 3.7, 3.8.

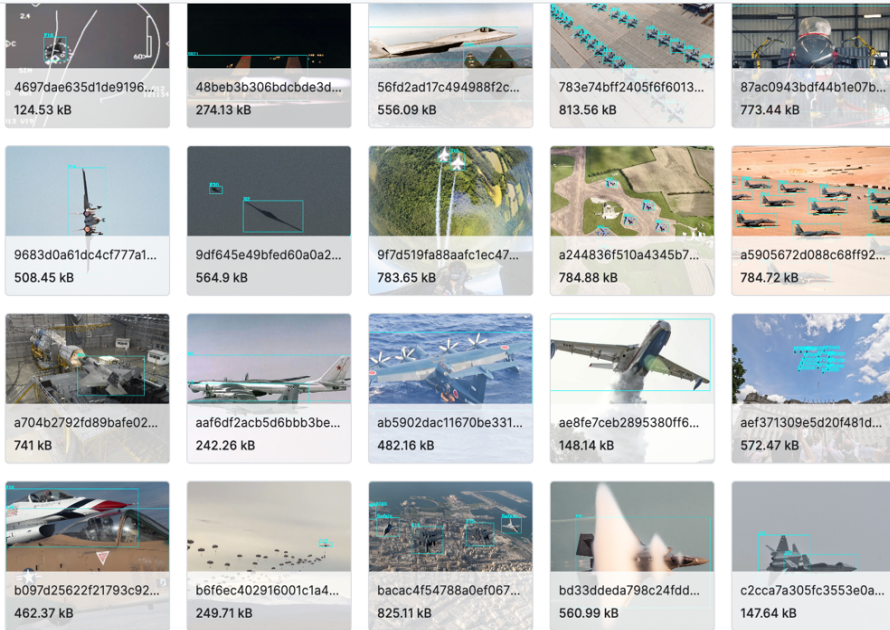
Military Aircraft Detection Dataset

Data Card Code (17) Discussion (2)

141 New Notebook Download (12 GB)

About this directory

sample images with bounding boxes



This preview shows 30 out of 40 items. [Load more](#)

- 02013d7acc296d5d1...
- 05e9819ea54482123r
- 154fef9572699a69b8
- 19e6b315b0709ea2b:
- 27ae5e3af2b84105c9
- 2e9f188061d9fb3288
- 30aaf62a22cd2dde0:
- 331a84b2b2b5d31a5-
- 3bb2fd94b28ac802c:
- 3c2af46161e0b1e839
- 4697dae635d1de919f
- 48beb3b306bdcbe3:
- 56fd2ad17c494988f2
- 783e74bff2405f6f601
- 87ac0943bdf44b1e07b...
- 9683d0a61dc4cf777a
- 9df645e49bfd60a0:
- 9fd519fa88afc1ec4
- a244836f510a4345b:
- a5905672d088c68ff9
- a704b2792fd89bafef0
- aaf6df2acb5d6bb3b:
- ab5902dac11670be331...
- ae8fe7ceb2895380ff6...
- aef371309e5d20f481d...
- ab5902dac11670be3:
- ae8fe7ceb2895380ff
- aef371309e5d20f481:

Summary

- 43.9k files
- 97.3k columns

Рисунок 3.7 – Набір даних на тему бойових літаків на ресурсі Kaggle, база фотографій

Military Aircraft Detection Dataset

Data Card Code (17) Discussion (2)

141 New Notebook Download (12 GB)

000aa01b25574f28b654718db0700f72.csv (313 B)

Detail Compact Column 8 of 8 columns

filename	# width	# height	class	# xmin	# ymin
1 unique value	4 total values	4 total values	JAS39 F35 Other (1)	50% 25% 25%	4 total values
000aa01b25574f28b654718db0700f72	2048	1365	F35	852	177
000aa01b25574f28b654718db0700f72	2048	1365	JAS39	169	769
000aa01b25574f28b654718db0700f72	2048	1365	JAS39	125	988
000aa01b25574f28b654718db0700f72	2048	1365	B52	277	901

No more data to show

Data Explorer

Version 56 (11.8 GB)

- annotated_samples
- crop
- dataset
 - 000106393cfe23438f
 - 000106393cfe23438f
 - 00032844ab679240f
 - 00032844ab679240f
 - 0003f56298fa899916
 - 0003f56298fa899916
 - 000aa01b25574f28b654718db0700f72
 - 000aa01b25574f28b654718db0700f72
 - 000e7662268a10718:
 - 000e7662268a10718:
 - 000ec980b5b17156a:
 - 000ec980b5b17156a:
 - 0036c2784a5b8b2a4
 - 0036c2784a5b8b2a4

Рисунок 3.8 – Набір даних на тему бойових літаків на ресурсі Kaggle, таблиця характеристик

Kaggle — платформа для змагань з аналітики та передбачувального моделювання, в рамках якого статистики та добувачі даних конкурують у створенні найкращих моделі для прогнозування та опису даних, запропонованих компаніями або користувачами. Цей краудсорсинговий підхід ґрунтується на тому, що є безліч стратегій, які можуть бути застосовані до будь-якого завдання з передбачувального моделювання, і наперед не відомо, яка методика або аналітичний підхід буде найбільш ефективним.

Під час пошуку даних для аналізу, також був знайдений приклад використання цих даних у вигляді базових функцій обробки [7].

3.6 Організація робочого середовища

Спершу, були намагання зібрати модель навчання в онлайн компіляторі, що надає сервіс Kaggle, але його потужностей виявилось недостатньо для швидкого аналізу. Навчання займало більше 12 годин, що робило неможливим роботу у цьому середовищі через розрив зв'язку з сервісом з причини спливу часу очікування.

Тому було проведено огляд онлайн рішень таких, як JetBrains Datalore, Google Colab, Paperspace Gradient. Це комерційні інструментарії, більш призначені для крупного бізнесу та їх тарифи розраховані на великі команди. Вони всі переслідують одну ціль: онлайн компіляція та навчання, робота зі збором моделей знань, редагування спеціалізованого коду.

Але монетизація цих рішень не підходила під формат “просто потестити”, та купляти підписку мало сенс маючи більш чіткі концепти.

Тому було прийняте рішення запуснути обробку набору даних локально на комп'ютері. Для цього, було використано середовище Jupyter notebook [3] – також це однойменна назва формату файлів, в яких і відбувається робота з набором даних, своєрідна огортка над Python кодом, структури JSON.

Проект Jupyter — це проект із розробки програмного забезпечення з відкритим кодом, відкритих стандартів і сервісів для інтерактивних обчислень,

використовуючи різні мови програмування. Назва проекту Jupyter є посиланням на три основні мови програмування, які підтримує Jupyter, а саме Julia, Python і R. Його назва та логотип є даниною поваги до відкриття Галілеєм супутників Юпітера, як це задокументовано в записниках, приписуваних Галілею. Проект Jupyter розробив і підтримував інтерактивні обчислювальні продукти Jupyter Notebook, JupyterHub і JupyterLab. Jupyter фінансово підтримується NumFOCUS.

Для розгортання середовища, знадобилось налаштувати потрібні бібліотеки Python, та слідувати документації JupyterLab, якихось додаткових несподіваних дій не знадобилось. Після цього, був запущений процес ініціалізації проекту, підтягування потрібних файлів та бібліотек, і після цього – навчання.

3.7 Реалізація навчання

Процес «навчання» та створення ML сутності реалізується програмним кодом, в більшості випадків через середовища, оглянуті в попередньому підрозділі. Поточний реалізуємий код розділений на тематичні блоки коду допоміжних функцій, що приймають участь у отриманні набору даних, підготовці їх у більш програмно читаємий формат та перевірку на помилки.

Набір функцій для створення ML сутності в даній ситуації є наступний:

- функції імпорту бібліотек, наприклад TensorFlow, Pandas, системні бібліотеки;
- функції для читання набору даних: посилання на папку в рамках файлової системи або хмари, валідація, перехоплення помилок;
- функції для створення дата фрейму (data frame [39]), тобто формату даних, який має розширені можливості по взаємодії з ним мовою Python, наприклад вкладені запити на дані через код, швидка проста обробка без написання додаткових великих функцій, тощо;
- функції, які перебирають дані, помічаючи їх валідність;

- функції, які відсіюють валідні дані;
- функції для виведення валідних даних для перевірки їх людиною;
- функції для оформлення історії навчання;
- конфігурація навчання;
- функція тренування.

Має сенс трохи розглянути конфігурацію навчання, детальний розбір лістингів коду може затягнутись надовго. Блок коду з конфігурацією представлений на рисунку 3.9.

```

: batch_size = 40 # set batch size for training
  epochs = 13 # number of all epochs in training, 40 by def
  patience = 1 # number of epochs to wait to adjust lr if monitored value does not improve
  stop_patience = 3 # number of epochs to wait before stopping training if monitored value does not improve
  threshold = 0.9 # if train accuracy is < threshold adjust monitor accuracy, else monitor validation loss
  factor = 0.5 # factor to reduce lr by
  ask_epoch = 5 # number of epochs to run before asking if you want to halt training
  batches = int(np.ceil(len(train_gen.labels) / batch_size)) # number of training batch to run per epoch

```

Рисунок 3.9 – Конфігурація навчання

Основні змінні в конфігурації:

- `batch_size` – розмір блоку даних на обробку. Існує, щоб обробляти дані послідовно, шматками, як метод протидії переповненню пам'яті;
- `epochs` – кількість циклів навчання. Концепція в тому, що кожний цикл навчання, машина в більшості випадків поліпшує знання про дані, на яких вона вчиться. Якщо привести аналогію з книгою, то це як перечитування книги людиною, чим більше разів людина перечитує її – тим більше фактів запам'ятовує та краще орієнтується у представленій інформації;
- `patience`, `stop_patience` – чутливість до припинення навчання: якщо код бачить, що умовно після 7 циклу, зміна якості знань не відбувається, а відбувається тільки корегування на 0.01%, код пропонує зупинити навчання, бо вірогідність, що можна буде отримати +25% до якості знань, як при перших 3 циклах, доволі мала.

Що стосується `epochs`, кількості циклів навчання, то спершу було обрано цифру за замовчанням, 40. Але після ряду запусків, було виявлено, що для навчання на поточному наборі даних, вистачить 12-14 циклів.

Історію навчання за 40 циклів з результатами в цифрах можна побачити на рисунку 3.10.

```

Do you want model asks you to halt the training [y/n] ?
n
Epoch   Loss   Accuracy  V_loss   V_acc   LR     Next LR  Monitor  % Improv  Duration
1 /40    8.296  30.124    5.28204  64.090  0.00100 0.00100 accuracy  0.00    1063.12
2 /40    3.984  70.580    2.72473  82.965  0.00100 0.00100 accuracy  134.30  1072.60
3 /40    2.060  86.532    1.59216  87.014  0.00100 0.00100 accuracy  22.60   1037.84
4 /40    1.216  92.605    1.12907  88.433  0.00100 0.00100 val_loss  29.09   1023.85
5 /40    0.844  95.833    0.93923  89.695  0.00100 0.00100 val_loss  16.81   1021.27
6 /40    0.666  97.489    0.82235  91.009  0.00100 0.00100 val_loss  12.44   1035.35
7 /40    0.566  98.127    0.76021  91.220  0.00100 0.00100 val_loss  7.56    1021.42
8 /40    0.493  98.600    0.72614  90.957  0.00100 0.00100 val_loss  4.48    1026.01
9 /40    0.445  98.896    0.67489  91.640  0.00100 0.00100 val_loss  7.06    1028.06
10 /40   0.422  98.810    0.63634  92.271  0.00100 0.00100 val_loss  5.71    1030.03
11 /40   0.375  99.349    0.63837  91.956  0.00100 0.00050 val_loss  -0.32   1021.25
12 /40   0.340  99.507    0.56301  92.587  0.00050 0.00050 val_loss  11.52   1019.26
13 /40   0.315  99.606    0.55217  92.955  0.00050 0.00050 val_loss  1.93    1043.39
14 /40   0.296  99.796    0.53257  92.797  0.00050 0.00050 val_loss  3.55    1016.66
15 /40   0.284  99.803    0.53417  92.744  0.00050 0.00025 val_loss  -0.30   1030.50
16 /40   0.271  99.829    0.51295  93.165  0.00025 0.00025 val_loss  3.68    1058.84
17 /40   0.263  99.901    0.51635  93.218  0.00025 0.00013 val_loss  -0.66   1045.17
18 /40   0.256  99.882    0.50798  93.375  0.00013 0.00013 val_loss  0.97    1039.90
19 /40   0.250  99.941    0.49927  93.428  0.00013 0.00013 val_loss  1.71    1025.91
20 /40   0.249  99.855    0.49396  93.586  0.00013 0.00013 val_loss  1.06    1044.35
21 /40   0.244  99.901    0.48892  93.428  0.00013 0.00013 val_loss  1.02    1022.51
22 /40   0.240  99.928    0.48436  93.218  0.00013 0.00013 val_loss  0.93    1024.68
23 /40   0.239  99.915    0.48402  93.481  0.00013 0.00013 val_loss  0.07    1045.94
24 /40   0.234  99.928    0.48317  93.323  0.00013 0.00013 val_loss  0.18    1056.51
25 /40   0.232  99.921    0.47775  93.428  0.00013 0.00013 val_loss  1.12    1019.79
26 /40   0.228  99.947    0.47246  93.691  0.00013 0.00013 val_loss  1.11    1018.29
27 /40   0.225  99.974    0.46612  93.796  0.00013 0.00013 val_loss  1.34    1046.27
28 /40   0.223  99.934    0.46856  93.691  0.00013 0.00006 val_loss  -0.52   1032.28
29 /40   0.220  99.954    0.46392  93.743  0.00006 0.00006 val_loss  0.47    1034.26
30 /40   0.218  99.954    0.46344  93.691  0.00006 0.00006 val_loss  0.10    1083.56
31 /40   0.218  99.908    0.46149  93.638  0.00006 0.00006 val_loss  0.42    1086.66
32 /40   0.216  99.941    0.45974  93.691  0.00006 0.00006 val_loss  0.38    1063.22
33 /40   0.216  99.941    0.45920  93.691  0.00006 0.00006 val_loss  0.12    1038.99
34 /40   0.214  99.941    0.46118  93.533  0.00006 0.00003 val_loss  -0.43   1064.07
35 /40   0.212  99.954    0.45817  93.638  0.00003 0.00003 val_loss  0.22    1042.73
36 /40   0.212  99.934    0.45470  93.533  0.00003 0.00003 val_loss  0.76    1085.72
37 /40   0.212  99.941    0.45606  93.481  0.00003 0.00002 val_loss  -0.30   1051.79
38 /40   0.210  99.961    0.45516  93.638  0.00002 0.00001 val_loss  -0.10   1050.70
39 /40   0.210  99.934    0.45223  93.743  0.00001 0.00001 val_loss  0.54    1054.98
40 /40   0.210  99.901    0.45325  93.638  0.00001 0.00000 val_loss  -0.23   1065.35
training elapsed time was 11.0 hours, 34.0 minutes, 54.12 seconds)

```

Рисунок 3.10 – Звіт навчання

Результатом такого навчання є можливість класифікації та виявлення об'єктів, в даному випадку повітряних, як зображено на рисунку 3.11.

Також важливим моментом є питання подання ПЗ до кінцевого пристрою та його завантаження у цей пристрій.

Встановлення ПЗ ІС на пристрій буде реалізуватись за допомогою Betaflight Configurator [27].



Рисунок 3.11 – Приклад класифікації та виявлення об'єкту

Betaflight Configurator – це кросплатформовий інструмент конфігурації для системи керування польотом Betaflight. Дозволяє налаштувати програмне забезпечення Betaflight для носіїв, а також встановлювати та оновлювати застосунки. Цей конфігуратор та сам Betaflight підтримуює велика кількість літаючої техніки, наприклад квадрокоптерів, гексакоптерів, октокоптерів та літаків з нерухомим крилом (fixed-wing aircraft).

Цей конфігуратор є єдиним конфігуратором, який підтримує спеціальні функції Betaflight. Таким чином, Betaflight Configurator – це програматор літальних об'єктів. Він дозволяє реалізовувати налаштування, аналіз та калібровку БПЛА.

4 РОЗРОБКА БІЗНЕС-АРХІТЕКТУРИ ІС ДЛЯ АНАЛІЗУ ПОВІТРЯНИХ ОБ'ЄКТІВ

4.1 Відомості про бізнес модель

Сама система для аналізу повітряних об'єктів працює за бізнес моделлю Solutions – це транзакційні стосунки де продукт або послуга розробляється та розповсюджується для користувачів з попередніми контактами та уточненнями з ними. Концепція бізнес-моделі представлена на рисунку 4.1.



Рисунок 4.1 – Концепція бізнес моделі рішень

Клієнти – військові та волонтерські формування.

Пропозиція створення цінності: допоміжне ПЗ для ІС БПЛА, яке автоматизує розпізнання та захоплення цілей.

Ланцюжок створення вартості:

- вартість за тестові моделі та супутні девайси для тестування коду;
- вартість за логістику будь-яких тестових девайсів;
- вартість за інструментарій для роботи команди, підписка на:
 - інструменти розробки (IDE, Git, тощо);
 - засоби комунікації;
 - засоби менеджменту процесу розробки та задач;
 - менеджери паролів та шифрування документів для безпечного зберігання та передачі чутливих даних;
 - інше;
- вартість за аренду та обслуговування офісів ;
- вартість за офісне обладнання;

- вартість за утримання персоналу;
- інше.

Для зображення системи для аналізу повітряних об'єктів та процесів, пов'язаних з нею, були обрані методи АТАМ та BPMN, а також схематизація бізнес-моделі.

4.2 Метод BPMN

Business Process Model and Notation (BPMN) — це графічне уявлення для визначення бізнес-процесів у моделі бізнес-процесу. Цей метод спочатку був розроблений Business Process Management Initiative (BPMI) та підтримується Object Management Group (OMG) після об'єднання цих двох організацій у 2005 році. Версія 2.0 BPMN була випущена в січні 2011 року, після чого було внесено зміни до моделі бізнес-процесу та нотації, щоб відобразити введення семантики виконання, яка була введена поряд з існуючими елементами нотації та діаграм. [31]

На рисунках 4.2-4.4 представлено три схеми що реалізують зображення трьох сценаріїв:

- захоплення, розрізнення та відстеження цілей: керований оператором літальний апарат рухається та відстежує цілі різних пріоритетності, маючи можливість утримувати увагу на цілі;
- доставка об'єкту на вказану ціль, або позицію/мітку: керований операором процес доставки ресурсу на вказану точку з додатковим аналізом місцевості та при наявності міток приземлення ресурсу – аналізу мітки;
- продовження відстеження цілі попри втрату зв'язку: при втраті радіозв'язку із літальним об'єктом, літальний об'єкт не знеточується, але втрачає зв'язок між оператором, при тому задача ПЗ з ІС в такому випадку – працювати як і раніше.

На рисунку 4.5 представлено перелік умовних позначень BPMN до рисунків 4.2-4.4.

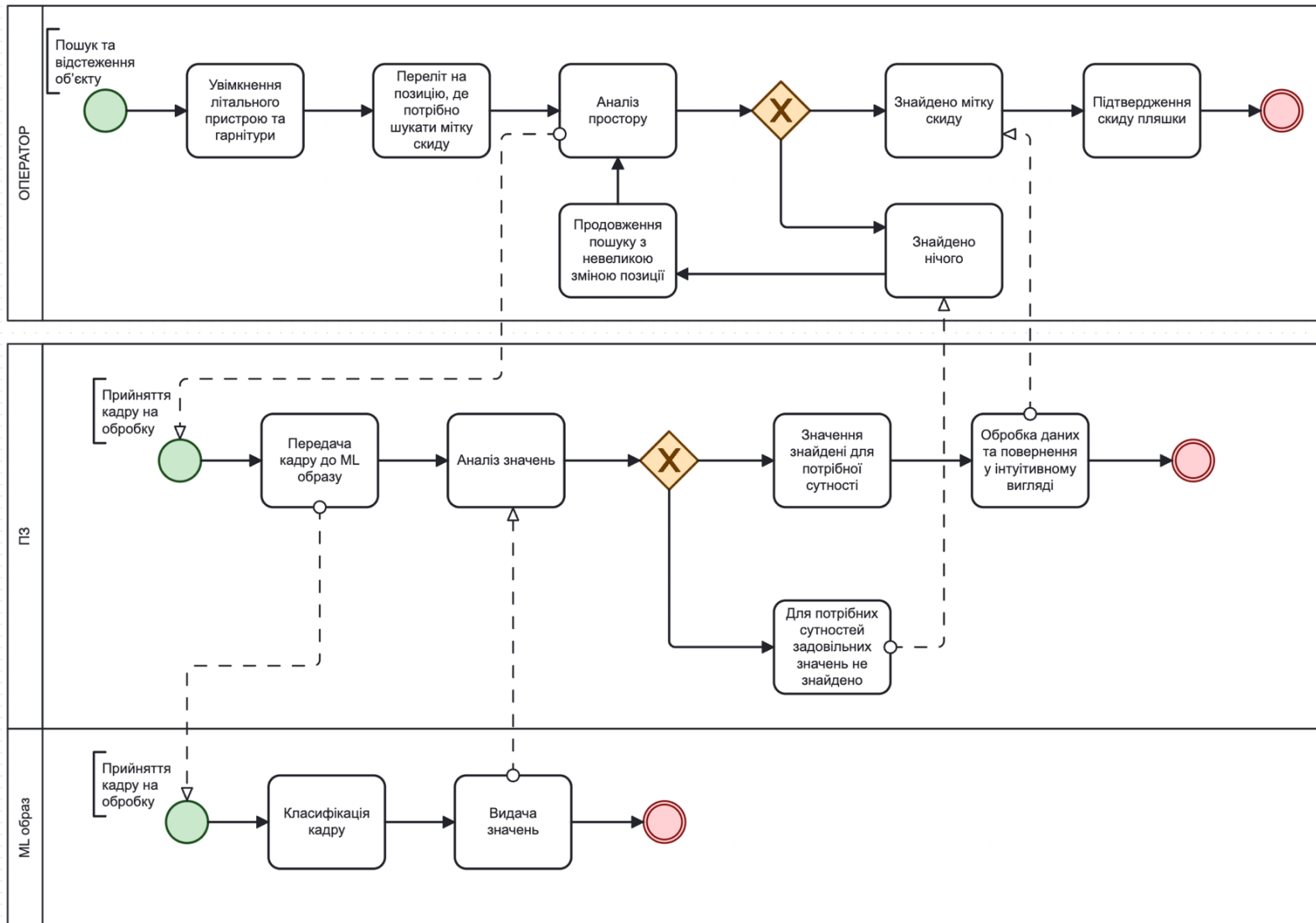


Рисунок 4.2 – Сценарій доставки об'єкту на вказану локацію

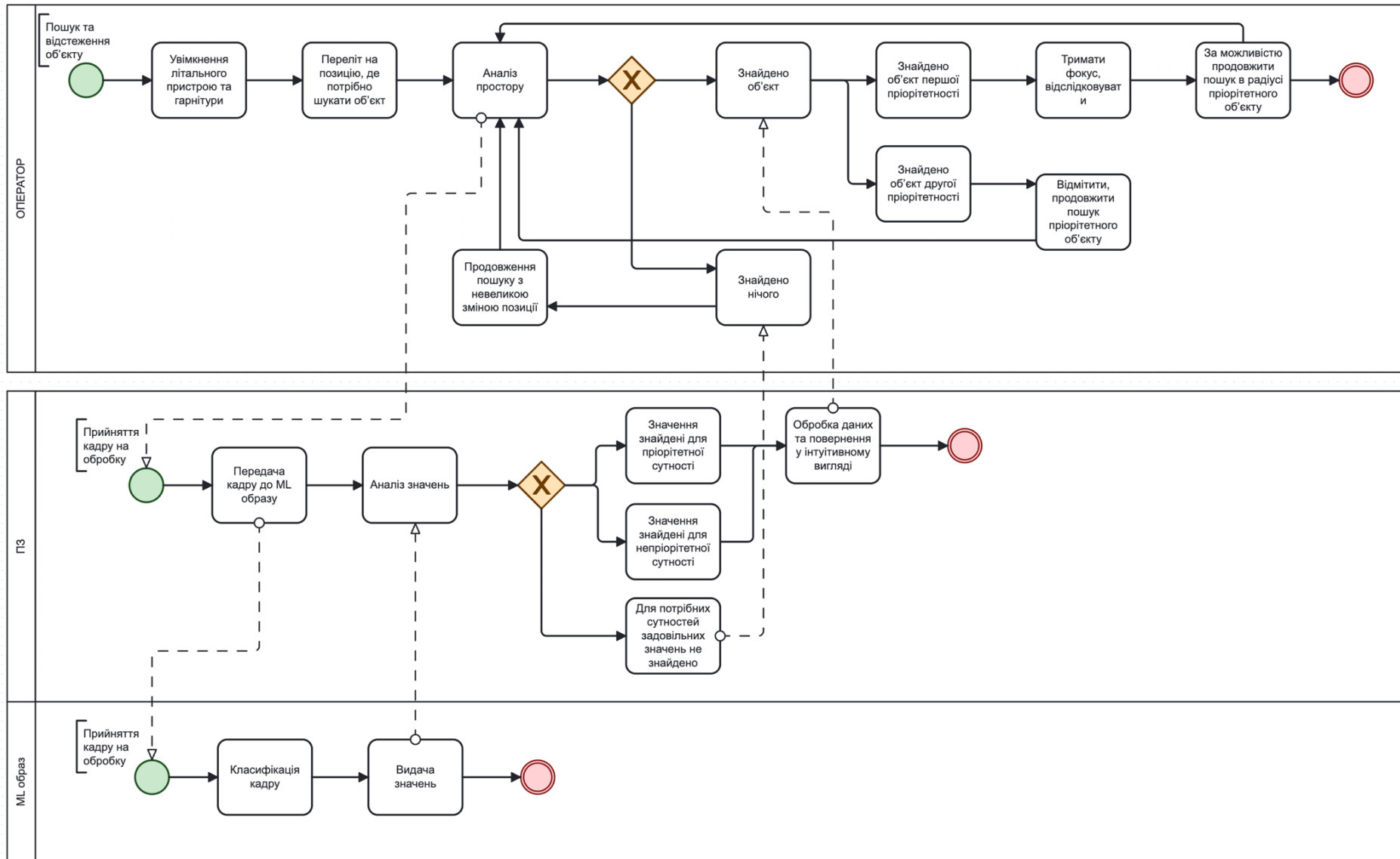


Рисунок 4.3 – Сценарій захоплення, розрізнення та відстеження цілей

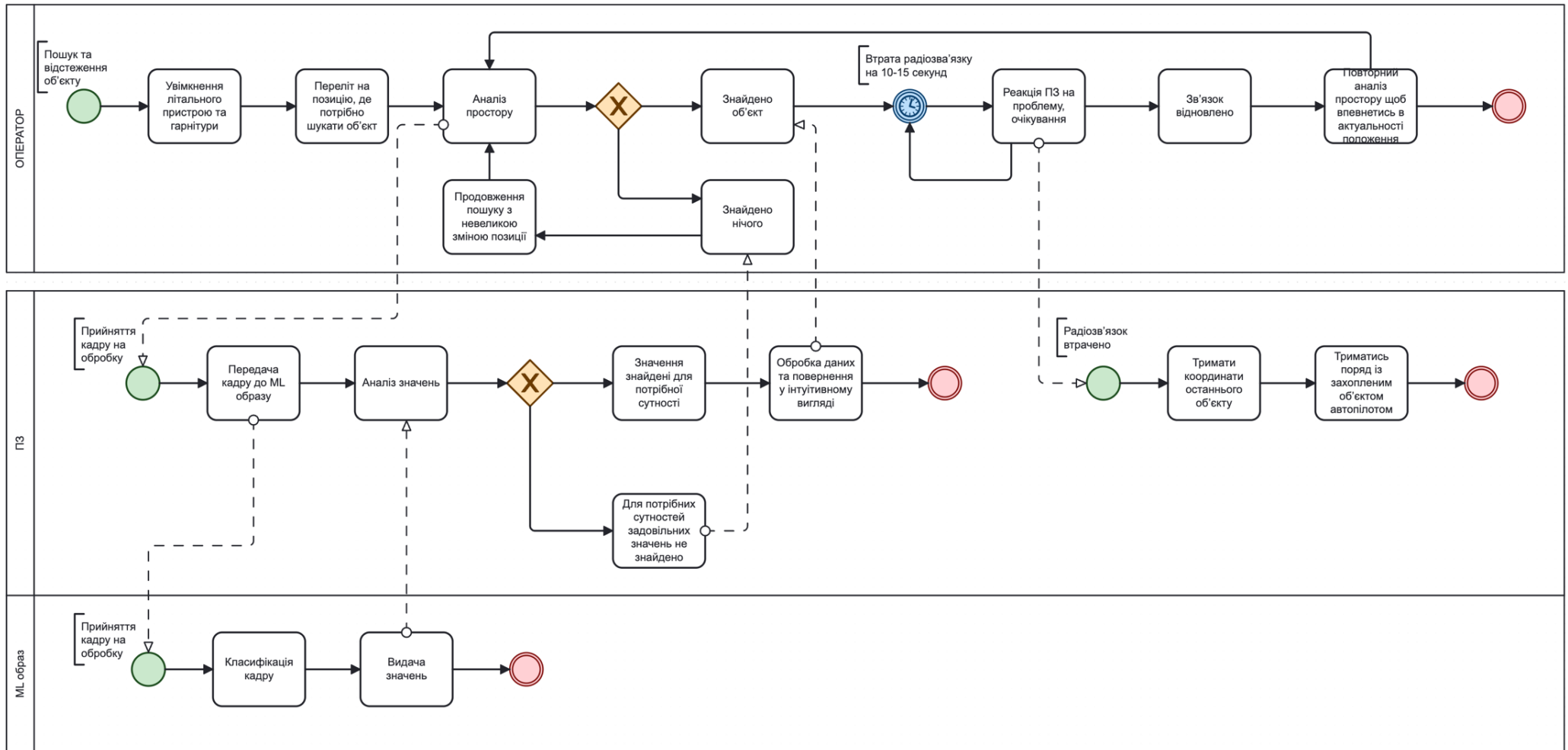


Рисунок 4.4 – Сценарій продовження відстеження цілі попри втрату зв'язку

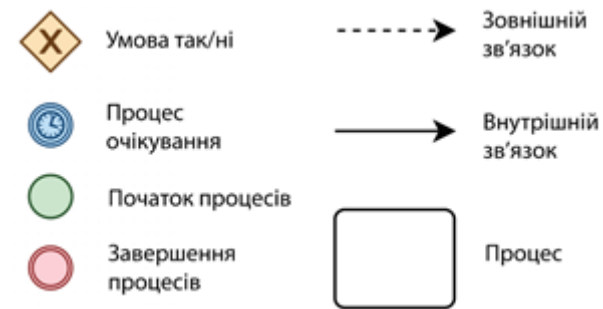


Рисунок 4.5 – Офіційний перелік умовних позначень BPMN

4.3 Метод АТАМ

Метод компромісного аналізу архітектури, Architecture tradeoff analysis method (АТАМ) — це процес зменшення ризику, який використовується на початку життєвого циклу розробки програмного забезпечення.

АТАМ був розроблений Інститутом розробки програмного забезпечення при Університеті Карнегі-Меллона. Його мета – допомогти вибрати відповідну архітектуру для програмної системи шляхом виявлення компромісів і чутливих моментів. [32]

4.3.1 Стейкхолдери системи

Стейкхолдери, або зацікавлена сторона – це сутність, метою якої є доведення проекту до певного етапу розвитку. Це може бути будь-хто в ланцюгу взаємодії із сервісом, починаючи від користувачів, закінчуючи замовником та/або інвестором.

Стейкхолдери проекту зображені на рисунку 4.6.

Клієнт – військова система, зацікавлена в підвищенні рейтингу ефективних влучань а також оптимізації втрат, збереження своєї живої сили.

Команда розробки – група технічних спеціалістів-ентузіастів, ціль якої – якісно реалізувати технічне завдання.

Посередники – волонтерські та комерційні організації, що будуть забезпечувати логістичне переміщення від місця розробки/тестування/складу.

Замовник – некомерційна організація, ціль якої – забезпечення ділянок країни де проводяться небажані для країни бойові дії, модифікованими літальними засобами в оборонних цілях.

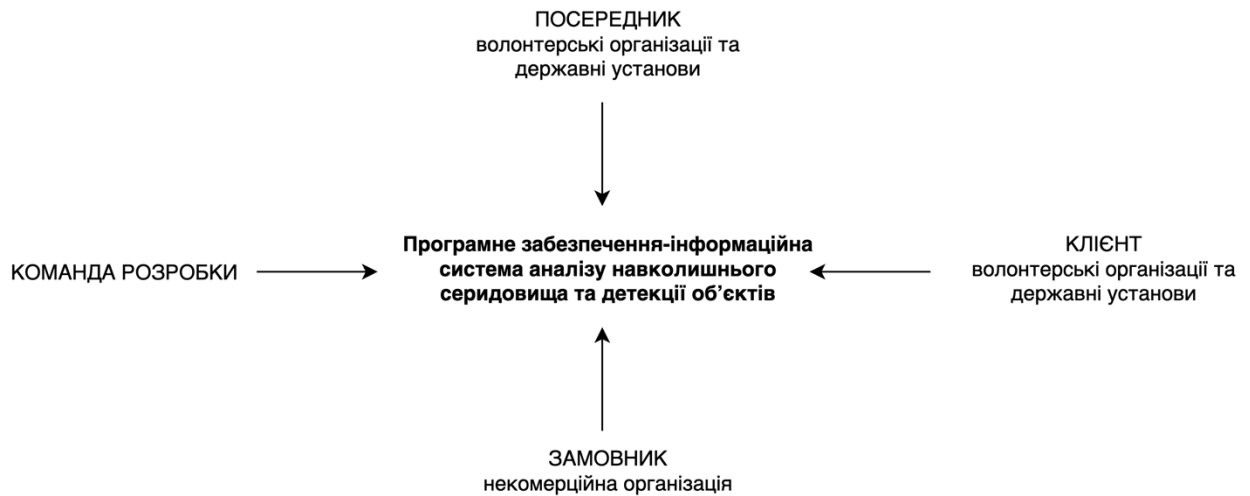


Рисунок 4.6 – Стейкхолдери проекту

4.3.2 Обмеження

Обмеженням та в тому числі мотивацією розробки проекту є періодична втрата контролю над літальним об'єктом через втрату радіосигналу через або прольот повз заважаючі об'єкти (проліт повз високого будинку), або через придушуючі сигнал дії техніки опонентів.

4.3.3 Атрибути якості

Атрибути якості – це пріоретизуємі метрики якості, за яких аналізується архітектурне рішення з різних боків.

Для системи для аналізу повітряних об'єктів притаманні базові та головні атрибути, такі як модифікуємість (modifiability), продуктивність (perfomance), доступність (availability), безпека (security) та багато іншого, що додається в залежності від пріоритетності питання (рисунок 4.7).

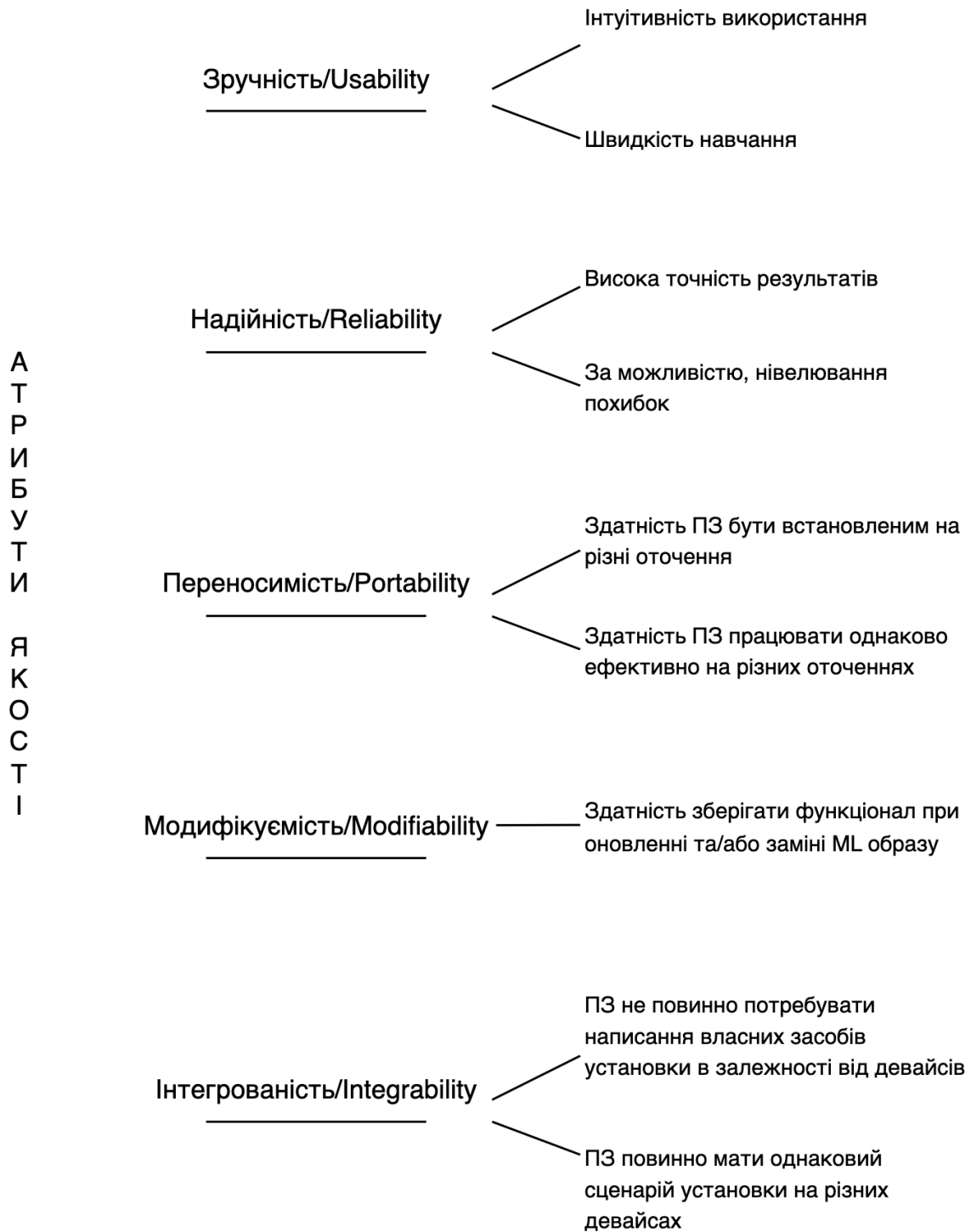


Рисунок 4.7 – Атрибути якості

Модифікуємість/Modifiability – ПЗ повинно бути “готовим” до, наприклад, заміни зібраної бази знань на оновлену, якщо наприклад, буде знайдено кращий набір даних для навчання, або з якихось причин буде зрозуміло що поточний навчений образ має вади.

Надійність/Reliability – Здатність ПЗ підтримувати визначену працездатність у заданих умовах. Максимально чітке визначення об’єктів, яке вийде реалізувати.

Зручність використання/Usability (або практичність) – Здатність ПЗ бути зручним у навчанні та використанні, а також привабливим для користувачів.

Переносимість/Portability – Здатність ПЗ зберігати працездатність при перенесенні з одного оточення в інше, включаючи організаційні, апаратні й програмні аспекти оточення.

Продуктивність/Performance – помилки в ПЗ можуть коштувати людського життя.

Інтегрованість/Integrability – процес інтеграції, доставки ПЗ з ІС до літального апарату повинен потребувати тільки популярного інструментарію на кшталт Betaflight Configurator та не повинен потребувати розробки додаткового ПЗ чисто заради “запису” в дрона.

Пріоритетність кожного атрибуту у контексті даного проекту показана в таблиці 4.1.

Таблиця 4.1 – Пріоритетність атрибутів

Атрибут	Важливість/Severity	Пріоритет/Priority
Модифікуємість/Modifiability	М	L
Надійність/Reliability	Н	Н
Зручність/Usability	Н	Н
Переносимість/Portability	М	L
Інтегрованість/Integrability	М	М
<i>L - low, M - Medium, H - High</i>		

ВИСНОВКИ

Дана кваліфікаційна робота призначена вирішувати питання допоміжних засобів класифікації та виявлення повітряних та наземних об'єктів за допомогою машинного навчання.

У першому та другому розділах кваліфікаційної роботи представлені основи машинного навчання та результати огляду використання машинного навчання та аналізу зображень у військовій справі.

У розділах три та чотири кваліфікаційної роботи представлені розробка технічної та бізнес-архітектури інформаційної системи для дослідження повітряних об'єктів.

Запропонована система, навчена на великому об'ємі інформації про повітряні цілі, дозволяє вирішити проблему виявлення цілей та їх класифікації. Але, в більшості випадків рекомендується використовувати цю технологію як допоміжну для оператора, а не заміну оператора. Крім того, особистістю системи є її програмна гнучкість.

В кваліфікаційній роботі використовувались тільки відкриті матеріали, робота не містить закритої, забороненої до публікації інформації.

За даною кваліфікаційною роботою було подано наукові тези на III Міжнародну науково-практичну конференцію «Intellectual education of students and schoolchildren of the new generation». Сертифікат закріплений в додатку Б.

ПЕРЕЛІК ПОСИЛАНЬ

1. Офіційна сторінка проекту kaggle [Електронний ресурс]
<https://www.kaggle.com> (дата звернення: 23.05.2023)
2. Офіційна сторінка проекту Jupyter [Електронний ресурс]
<https://jupyter.org> (дата звернення: 23.05.2023)
3. Project Jupyter [Електронний ресурс]
https://en.wikipedia.org/wiki/Project_Jupyter (дата звернення: 23.05.2023)
4. Top 10 Python Packages for Machine Learning [Електронний ресурс]
<https://www.activestate.com/blog/top-10-python-machine-learning-packages> (дата звернення: 23.05.2023)
5. Best Python libraries for Machine Learning [Електронний ресурс]
<https://www.geeksforgeeks.org/best-python-libraries-for-machine-learning> (дата звернення: 23.05.2023)
6. Military Aircraft Detection Dataset [Електронний ресурс]
<https://www.kaggle.com/datasets/a2015003713/militaryaircraftdetectiondataset>
(дата звернення: 23.05.2023)
7. Military Aircraft Detection | EfficientNetB3 | 93% [Електронний ресурс]
<https://www.kaggle.com/code/abdallahwagih/military-aircraft-detection-efficientnetb3-93> (дата звернення: 23.05.2023)
8. Офіційна сторінка проекту Deepstatemap [Електронний ресурс]
<https://deepstatemap.live> (дата звернення: 23.05.2023)
9. Flask API for Machine Learning Model from scratch [Електронний ресурс]
<https://gaurav-patil21.medium.com/flask-api-for-machine-learning-model-from-scratch-d2456ef34bcf> (дата звернення: 23.05.2023)
10. Build ML models as best API and test with thunder client [Електронний ресурс]
<https://github.com/amitvkulkarni/Build-ML-Models-As-Rest-API> (дата звернення: 23.05.2023)
11. Automating data analytics with ChatGPT [Електронний ресурс]
<https://medium.com/data-science-at-microsoft/automating-data-analytics-with->

chatgpt-827a51eaa2c (дата звернення: 23.05.2023)

12. 7 Key Military Applications of Machine Learning [Електронний ресурс] <https://medium.com/@nqabell89/7-key-military-applications-of-machine-learning-9818dfa2ea86> (дата звернення: 23.05.2023)

13. Дрон зі штучним інтелектом вже виконує бойові завдання в ЗСУ [Електронний ресурс] <https://armyinform.com.ua/2023/09/04/dron-zi-shtuchnym-intelektom-vzhe-vykonuye-bojovi-zavdannya-v-zsu> (дата звернення: 23.05.2023)

14. Artificial intelligence (AI) vs. machine learning (ML) [Електронний ресурс] <https://cloud.google.com/learn/artificial-intelligence-vs-machine-learning> (дата звернення: 23.05.2023)

15. Valkyrie: This Autonomous AI Drone Could Be the Military's Next Weapon | WSJ Equipped [Електронний ресурс] <https://www.youtube.com/watch?v=HCjlokSzIKY> (дата звернення: 23.05.2023)

16. 12 Best Jupyter Notebook Alternatives [2023] – Features, pros & cons, pricing [Електронний ресурс] <https://noteable.io/blog/jupyter-notebook-alternatives> (дата звернення: 23.05.2023)

17. Current and Future Applications of Machine Learning for the US Army [Електронний ресурс] <https://apps.dtic.mil/sti/pdfs/AD1050263.pdf> (дата звернення: 23.05.2023)

18. Design an application of image processing in military [Електронний ресурс] <https://medium.com/@sakshi.dond19/design-an-application-of-image-processing-in-military-f409656de878> (дата звернення: 23.05.2023)

19. Video processing image processing artificial intelligence machine learning [Електронний ресурс] <https://www.militaryaerospace.com/computers/article/14289480/video-processing-image-processing-artificial-intelligence-machine-learning> (дата звернення: 23.05.2023)

20. Harnessing Communities of Knowledge: Building an Automated Species Identification Tool [Електронний ресурс] https://www.youtube.com/watch?v=Y_9vj7f9WU4 (дата звернення: 23.05.2023)

21. Automatic Number Plate Recognition (ANPR) Technology [Електронний ресурс] https://www.youtube.com/watch?v=CJ__63nn488 (дата звернення: 23.05.2023)
22. Seeing like a machine: a beginners guide to image analysis in machine learning [Електронний ресурс] <https://www.datacamp.com/tutorial/seeing-like-a-machine-a-beginners-guide-to-image-analysis-in-machine-learning> (дата звернення: 23.05.2023)
23. Image analysis [Електронний ресурс] https://en.wikipedia.org/wiki/Image_analysis (дата звернення: 23.05.2023)
24. Бізнес моделі [Електронний ресурс] <https://www.businessmodelzoo.com> (дата звернення: 23.05.2023)
25. Якість програмного забезпечення [Електронний ресурс] https://uk.wikipedia.org/wiki/Якість_програмного_забезпечення (дата звернення: 23.05.2023)
26. Офіційна сторінка проекту Betaflight [Електронний ресурс] <https://betaflight.com> (дата звернення: 23.05.2023)
27. How to Setup Betaflight Firmware [Електронний ресурс] <https://oscarliang.com/betaflight-firmware-setup> (дата звернення: 23.05.2023)
28. Business Process Model and Notation [Електронний ресурс] https://en.wikipedia.org/wiki/Business_Process_Model_and_Notation (дата звернення: 23.05.2023)
29. Architecture tradeoff analysis method [Електронний ресурс] https://en.wikipedia.org/wiki/Architecture_tradeoff_analysis_method (дата звернення: 23.05.2023)
30. Офіційна сторінка проекту Griselda [Електронний ресурс] <https://www.griselda.com.ua> (дата звернення: 23.05.2023)
31. Офіційна сторінка проекту Saker [Електронний ресурс] <https://saker.airforce> (дата звернення: 23.05.2023)
32. Офіційна сторінка проекту Shield AI [Електронний ресурс] <https://shield.ai> (дата звернення: 23.05.2023)

33. Офіційна сторінка проекту Chat GPT [Електроний ресурс]
<https://chat.openai.com> (дата звернення: 23.05.2023)
34. Офіційна сторінка проекту Neptune [Електроний ресурс]
<https://neptune.ai> (дата звернення: 23.05.2023)
35. Military Aircraft Detection Dataset [Електроний ресурс]
<https://datasetninja.com/military-aircraft-detection> (дата звернення: 23.05.2023)
36. Total military aircraft strength in the Asia-Pacific region in 2023, by country or territory [Електроний ресурс]
<https://www.statista.com/statistics/934070/asia-pacific-military-aircraft-strength-by-country> (дата звернення: 23.05.2023)
37. 2022 Russia Ukraine War [Електроний ресурс]
<https://www.kaggle.com/datasets/piterfm/2022-ukraine-russian-war/data> (дата звернення: 23.05.2023)
38. The pandas DataFrame: Make Working With Data Delightful
39. Tsopa O. Basic technologies and techniques ml/ai for improving physical layer security for 5g/6g communications systems / O. Tsopa, O. Dudka A. Merzlikin // System analysis and intelligent systems for management : The 17th International scientific and practical conference, Ankara, Turkey, 2-5 May 2023. – P. 403–408.
40. [Електроний ресурс] <https://realpython.com/pandas-dataframe/#:~:text=The%20pandas%20DataFrame%20is%20a,with%20in%20Excel%20or%20Calc.> (дата звернення: 23.05.2023)