

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА  
Факультет прикладної математики та інформатики  
Кафедра прикладної математики

## Дипломна робота

РОЗПІЗНАВАННЯ ОБ'ЄКТІВ НА АЕРОЗНІМАЛЬНИХ ЗОБРАЖЕННЯХ

Виконала: студентка групи ПМП-42  
спеціальності

113 – прикладна математика

Яловега Т. Б.

Керівник Дяконюк Л. М.

Рецензент \_\_\_\_\_

Львів 2023

## ЗМІСТ

<b>ВСТУП</b> .....	3
<b>РОЗДІЛ 1</b> .....	5
<b>ПОСТАНОВКА ЗАДАЧІ</b> .....	5
<b>1.1 Оцінка сучасного стану проблеми</b> .....	5
<b>1.2 Актуальність теми</b> .....	6
<b>1.3 Формулювання задачі</b> .....	7
<b>РОЗДІЛ 2</b> .....	9
<b>ТЕОРЕТИЧНІ ВІДОМОСТІ З ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОБОТИ З ДАНИМИ, ПОДАНИМИ У ВИГЛЯДІ ЗОБРАЖЕНЬ</b> .....	9
<b>2.1 Подання зображень</b> .....	9
<b>2.2 Класифікації зображень за допомогою згорткової нейронної мережі</b> .....	11
<b>2.3 Бібліотеки, що використовуються для розв’язання задач розпізнавання об’єктів</b> .....	15
<b>РОЗДІЛ 3</b> .....	17
<b>ЗАСТОВУВАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ВІЙСЬКОВИХ ЛІТАКІВ</b> .....	17
<b>3.1 Підготовка даних</b> .....	17
<b>3.2 Побудова моделі</b> .....	19
<b>3.3 Модель для виявлення декількох об’єктів на фото</b> .....	21
<b>ВИСНОВКИ</b> .....	24
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ</b> .....	25
<b>ДОДАТОК А</b> .....	27

## ВСТУП

Розпізнавання образів – це галузь штучного інтелекту, яка вивчає методи класифікації об'єктів. Зазвичай об'єкт, який піддається класифікації, називається образом. Образами можуть бути цифрові зображення, символи або звукові записи.

Ця галузь входить в загальну область машинного навчання, яка дозволяє комп'ютерам навчатись на основі вхідних даних і будувати моделі без явного програмування.

Розпізнавання зображень з використанням штучного інтелекту є давньою проблемою в галузі комп'ютерного зору. Хоча існує багато різних методів, загальна мета полягає у класифікації об'єктів на основі їх зображень та віднесенні їх до різних категорій.

Для розпізнавання образів на зображеннях застосовуються різні методики. Наприклад, використовуються методи, які базуються на знаходженні контуру об'єкта та вивченні його властивостей. Також важливим підходом є аналіз об'єктів з різних масштабів, кутів та зсувів.

Однак, у цій роботі ми зосередимося на використанні штучних нейронних мереж для розпізнавання образів на зображеннях.

Штучні нейронні мережі є важливим інструментом в області розпізнавання образів. Вони моделюють процеси, схожі на ті, що відбуваються у мозку людини, дозволяючи комп'ютерам виконувати складні завдання розпізнавання та обробки зображень. Зокрема, ми зосередимося на згорткових нейронних мережах, які є одним з найпоширеніших типів штучних нейронних мереж для розпізнавання об'єктів на зображеннях. Особливістю цього типу мереж є згорткові шари.

Вивчення цього підходу має велике значення, оскільки розпізнавання образів знаходить широке застосування в різних сферах, включаючи комп'ютерний зір, медичну діагностику, автономні системи та безпеку. Розуміння принципів та ефективного використання згорткових нейронних мереж допоможе вирішувати складні завдання розпізнавання образів з високою точністю та швидкістю.

Розглянемо проблему розпізнавання військових літаків на зображеннях із супутника та розподілу виявлених об'єктів на класи.

**Мета роботи.** Створення згорткової нейронної мережі для розпізнавання та розподілу військових літаків на класи в залежності від моделі літака. Вхідними даними для моделі є аерознімки із супутників. Для досягнення мети вирішуються наступні завдання:

- Огляд різних підходів до представлення та обробки зображень
- Вивчення застосування методів машинного навчання для розпізнавання образів на фотографіях з супутників
- Аналіз існуючих методів розв'язання даної проблеми
- Вибір набору даних для навчання моделі
- Розробка нейронної мережі для виявлення образів і класифікації
- Оцінка отриманих результатів

## РОЗДІЛ 1

### ПОСТАНОВКА ЗАДАЧІ

#### 1.1 Оцінка сучасного стану проблеми

Останні досягнення в галузі машинного навчання показують значні успіхи в галузі комп'ютерного зору. Зокрема, методи глибокого навчання для розпізнавання зображень продемонстрували високу продуктивність та гнучкість. Ми бачимо на ринку багато прикладів програм, де використовуються технології комп'ютерного зору. Наведемо деякі з них:

- AirSense - функція, розроблена провідним виробником дронів DJI. Він використовує алгоритми машинного навчання, щоб дозволити дронам DJI виявляти пілотовані літаки під час польоту і знижувати ризики зіткнень.
- Inspektlabs – застосунок, який за допомогою фотографій чи відео транспортних засобів виявляє та класифікує їх дефекти, таких як подряпина, тріщина, корозія і т.д.
- Assaia – використовує комп'ютерний зір для відстеження повітряних і наземних транспортних засобів для подальшої оптимізації руху літаків.

Згорткові нейронні мережі широко застосовуються в завданнях обробки зображень. Проте для досягнення високої точності зазвичай потрібна велика кількість шарів та параметрів, що може призвести до значного збільшення обчислювальних витрат. Це ускладнює навчання та розгортання моделей на обмежених обчислювальних ресурсах. Однак, існують підходи до створення легких згорткових нейронних мереж, які мають меншу глибину, але ще можуть бути вдосконалені для поліпшення точності розпізнавання.

## 1.2 Актуальність теми

Тема дослідження є особливо актуальною під час російсько-української війни. У контексті воєнних дій, досить точне та швидке розпізнавання ворожих повітряних суден може мати вирішальне значення для стратегічного планування та оборони. Застосування методів комп'ютерного зору дозволяє автоматизувати процес виявлення та класифікації ворожих літаків, що, у свою чергу, допомагає пришвидшити процеси прийняття рішень та ефективніше їм протидіяти.

Також розпізнавання об'єктів на аерознімках може використовуватись у наведенні ракет та інших бортових систем озброєння. Відстежуючи положення та переміщення цілей, комп'ютерне бачення може гарантувати ураження ракетами своїх цілей з достатньо високою точністю. Ця технологія також використовується для розробки автономних систем озброєння, які можуть функціонувати з обмеженою участю людини.

Автономні системи наведення здатні на основі комп'ютерного зору розрізняти цілі та постійно оновлювати дані про траєкторію польоту, що може збільшити точність ураження.

Отже, дослідження методів комп'ютерного зору є корисними як для воєнної розвідки, так і для наведення на цілі.

Виявлення образів все ще активно вивчається та вдосконалюється, адже останнім часом спостерігається збільшення обсягів інформації, яка зберігається, передається та обробляється в цифровому вигляді. Це пов'язано з все більшим проникненням цифрових технологій в науку, техніку і повсякденне життя.

У зв'язку зі зростанням обсягу цифрової інформації і розширенням застосувань цифрових технологій, дослідження, спрямовані на створення ефективних нейронних мереж і розробку методів навчання для обробки та

класифікації зображень, залишаються актуальними. Крім цього, необхідно досліджувати методи, що сприятимуть прискоренню процесу навчання моделей, а також розробляти різноманітні підходи для поліпшення їх ефективності.

### 1.3 Формулювання задачі

Розглянемо задачу розпізнавання військових повітряних суден на аерознімках. Нехай маємо зображення із супутника, приклад якого поданий на рисунку 1.1. Завданням є виявити на ньому літаки та класифікувати їх.



Рис. 1.1 – Приклад вхідних даних.

Отже, задача полягає у створенні згорткової нейронної мережі, яка буде класифікувати літаки різних моделей. А також для використання цієї мережі на фотографіях, де міститься декілька літаків, потрібно застосувати методи виявлення об'єктів на зображенні, або іншими словами, локалізації області, де міститься один літак.



## РОЗДІЛ 2

# ТЕОРЕТИЧНІ ВІДОМОСТІ З ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОБОТИ З ДАНИМИ, ПОДАНИМИ У ВИГЛЯДІ ЗОБРАЖЕНЬ

### 2.1 Подання зображень

У сучасному світі значна частина інформації представлена у вигляді графічних зображень. Відтак обробка зображень має велике значення в галузі машинного навчання, яка спеціалізується на аналізі та опрацюванні інформації.

Цифрове зображення може бути представлено як двовимірна функція  $f(x, y)$ , що є проекцією тривимірної сцени на площину. Координати  $x$  та  $y$  відповідають розташуванню пікселів, а значення функції вказують на їх інтенсивність. З математичної точки зору, це представляється двовимірною матрицею, де кожний елемент матриці відповідає кольору пікселя.

У цифровому представленні зображень широко використовуються різні кольорові моделі. Основні з них такі:

- Модель відтінків сірого. Вона використовує різні відтінки сірого для представлення кольору. Кожен відтінок в діапазоні від 0 до 255 відповідає певному колірному значенню, де 0 позначає білий, а 255 - чорний. Проміжні значення представляють відтінки, які формуються шляхом змішування кольорів.
- Модель RGB. Ця модель базується на трьох основних кольорах: червоному (red), зеленому (green) і синьому (blue). Колір формується шляхом накладання цих трьох кольорів один на

одного на чорному фоні. В моделі RGB кожен колір представлений значенням його інтенсивності.

- Модель HSI. Вона представляє колір як комбінацію відтінку (hue), насиченості (saturation) і інтенсивності (intensity). Вона надає більш широкі можливості для опису кольорів, порівняно з моделлю RGB. Модель HSI особливо корисна для алгоритмів обробки зображень.

Ці кольорові моделі дозволяють працювати з зображеннями в контексті алгоритмів машинного навчання, оскільки вони можуть бути представлені числовими значеннями, з якими ці алгоритми працюють.

Підготовка зображень перед навчанням моделі є важливим кроком для поліпшення результатів. Деякі методи обробки зображень включають:

- Зміна кольору: Зображення можуть бути перетворені з однієї кольорової моделі, наприклад, з RGB до чорно-білої, що може сприяти спрощенню обробки та зменшенню розміру даних.
- Зменшення шуму: Прибирання непотрібних деталей, які можуть пошкодити якість зображення, може позитивно вплинути на подальший аналіз.
- Урізноманітнення та розширення даних: Для збільшення обсягу навчальних даних можна застосовувати різноманітні перетворення, такі як віддзеркалювання, обертання, обрізання та інші.

Ці методи оптимізації сприяють покращенню вхідних даних для тренування моделі, особливо у випадку обмеженої кількості навчальних прикладів або потреби у більшій різноманітності даних.

## 2.2 Класифікації зображень за допомогою згорткової нейронної мережі

Згорткова нейронна мережа (CNN - Convolutional Neural Networks) є специфічним типом нейронних мереж, часто використовуваним для розпізнавання зображень. Вона має спеціальний шар згортки, який дозволяє зменшити розмірність зображень, не втрачаючи при цьому інформацію. Це робить CNN особливо ефективною для завдань з обробки зображень.

Класифікація зображень включає виділення характеристик з зображень для виявлення закономірностей в наборі даних. Використання стандартних штучних нейронних мереж (ANN - artificial neural networks) для класифікації зображень може бути обчислювально витратним, оскільки вимагає великої кількості нейронів.

З цієї причини згорткова нейронна мережа використовує інший підхід, симулюючи спосіб, яким ми сприймаємо навколишнє середовище з допомогою очей. При перегляді зображень ми автоматично ділимо їх на багато малих підзображень і аналізуємо їх поодиноці. Ці підзображення обробляються та інтерпретуються, що дозволяє нам аналізувати та розуміти зображення. Розглянемо, як цей принцип реалізований у згортковій нейронній мережі.

У згортковому шарі ми працюємо з фільтром або ядром, які проходять по матриці зображення. Фільтр накладається на матрицю таким чином, щоб його верхній лівий кут збігався з відповідним кутом матриці зображення. Здійснюється елементне перемноження фільтра з елементами матриці зображення, які він покриває. Після цього обчислюється сума всіх добутків і записується як перший елемент в нову матрицю. Потім з певним кроком фільтр зсувається вправо по матриці зображення, і алгоритм повторюється для кожного фрагмента. Візуалізація цих операцій наведена на рисунку 2.1.

Цей крок дозволяє зменшити розмірність зображення.

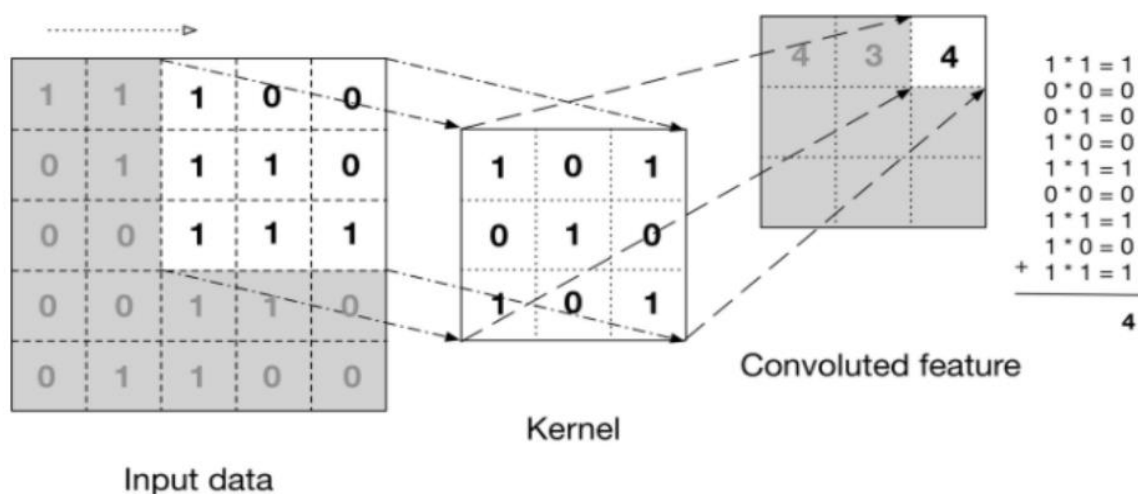


Рис. 2.1 – Приклад операції згортки

На наступному етапі відбувається об'єднання, цей шар також відомий як "Pooling". Цей процес подібний до згорткового шару, але використовує математичну функцію замість лінійних перетворень. Найпоширеніші функції – це Max Pooling і Average Pooling.

Під час проходження Max Pooling по матриці повертається максимальне значення з кожного фрагмента, на який цей метод застосовується. Average Pooling натомість обчислює середнє значення. Зазвичай Max Pooling вважається більш ефективним, оскільки він також допомагає знизити шум при обході матриці, а не лише зменшити її розмірність.

На рисунку 2.2 подано приклад операції об'єднання

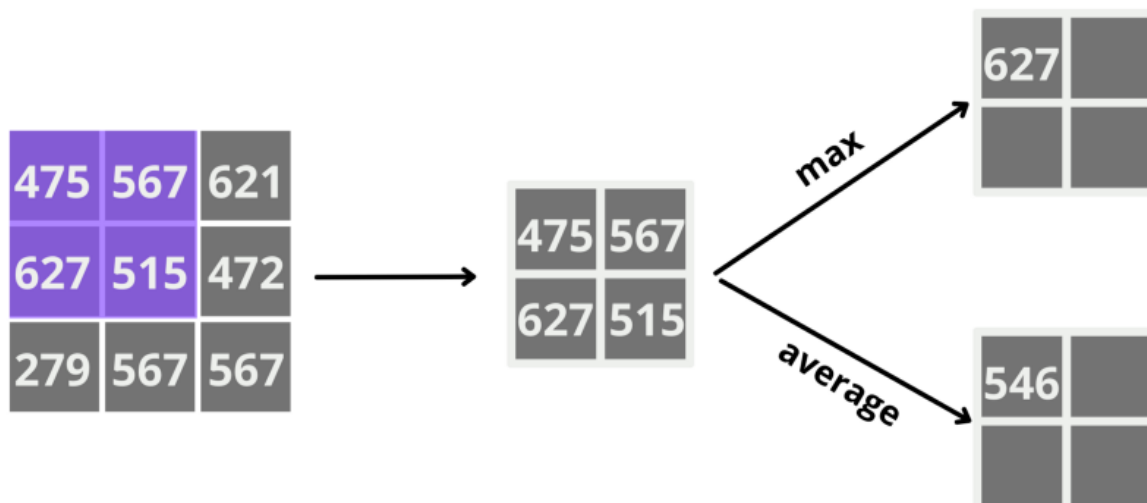


Рисунок 2.2 – Виконання операції об'єднання

Застосування об'єднуючого шару після згортки утворює згортковий шар у нейронній мережі.

У задачах класифікації зображень після виконання операцій згортки і об'єднання матриці, отримані з попередніх шарів, перетворюються на впорядкований список значень, де кожен елемент списку відповідає певному елементу матриці. Після цього до цього списку застосовуються звичайні штучні нейронні мережі.

Тому наступним етапом є повністю зв'язаний шар, який використовується у звичайних нейронних мережах. Ми можемо використовувати повністю зв'язні шари, адже значно зменшили зображення на попередніх шарах. Тут окремі фрагменти зображення знову з'єднуються, щоб встановити зв'язки та класифікувати знімок.

На вихідному шарі створюють таку кількість нейронів, скільки було класів у зображеннях. На рисунку 2.3 можна побачити нейронні зв'язки, на цьому шарі.

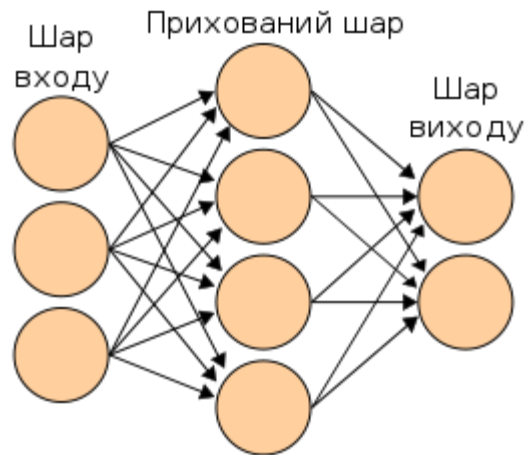


Рис 2.3 – Схема штучної нейронної мережі

Кожен зв'язок між нейронами має власну вагу. Ці ваги міняються в процесі навчання вдосконалюючи цим нейронну мережу. Вона може використовувати ваги повторно чи зменшувати кількість параметрів для кращої взаємодії з отриманими даними

Кожен шар моделі використовує активаційну функцію для обчислення вихідного значення в кожному нейроні. Вибір різних активаційних функцій призводить до отримання різних результатів, що дає можливість налаштувати модель. Таким чином, відповідний вибір активаційних функцій допомагає покращити передбачення моделі[4].

- ReLU перетворює від'ємні значення вхідних даних на нуль, залишаючи додатні значення без змін. Ця функція є простою та швидкою. Зазвичай вона використовується в прихованих шарах нейронних мереж.
- Функція Softmax часто використовується в вихідному шарі нейронної мережі та демонструє хороші результати в багатокласовій класифікації.
- Функція Sigmoid перетворює значення вхідних даних з проміжку  $(-\infty; +\infty)$  на проміжок  $[0; 1]$ . Вона не рекомендується для регресійних задач,

оскільки може призвести до зниження точності моделі та вимагає багато часу під час навчання.

Отже, для класифікації зображень доцільно використовувати згорткову нейронну мережу, адже її згортковий шар зменшує високі розмірності зображень без втрати інформації. Це ефективний і часто використовуваний метод розв'язання подібних задач.

### **2.3 Бібліотеки, що використовуються для розв'язання задач розпізнавання об'єктів**

Розв'язувати задачу розпізнавання повітряних суден будемо за допомогою згорткових нейронних мереж, які ефективно працюють із зображеннями.

У цьому випадку ми використовуватимемо бібліотеку Tensorflow, яка є зручним і відкритим інструментом для чисельного програмування на мові Python. Вона спрощує процес машинного навчання, роблячи його швидшим та простішим. Python надає зручний спосіб поєднувати високорівневі абстракції разом.

У Tensorflow вузли та тензори є об'єктами Python, а програми Tensorflow виконуються як звичайні Python-додатки. Ця бібліотека може навчати та запускати глибокі нейронні мережі для різних завдань, зокрема для розпізнавання зображень.

Один з ключових концепцій Tensorflow – це графи потоків даних. Вони представляють структури, що описують, як дані переміщуються через граф або послідовність обчислювальних вузлів. Кожен вузол на графі виконує математичну операцію, а з'єднання між вузлами представлені багатовимірними масивами даних (чи тензорами).

Keras – це високорівневий інтерфейс для машинного навчання, призначений для спрощення процесу розробки і навчання глибоких нейронних мереж. Він працює поверх інших бібліотек машинного навчання, таких як Tensorflow, і надає простий у використанні API для визначення, тренування та оцінки моделей машинного навчання.

Keras пропонує широкий спектр вбудованих функцій та модулів, що дозволяють створювати різноманітні типи моделей, включаючи послідовні моделі, моделі з функціональним API та моделі з використанням графів. Він також підтримує різні шари, функції активації, функції втрат, оптимізатори та інші компоненти, що використовуються для налаштування та навчання нейронних мереж.

Keras є потужним інструментом для розв'язання задач машинного навчання і здобув популярність завдяки своїй простоті використання і гнучкості. Він часто використовується як один з найпопулярніших інтерфейсів для розробки глибоких нейронних мереж.

YOLOv5 виконує виявлення об'єктів, розділяючи вхідне зображення на сітку комірок та передбачаючи обмежувальні рамки та ймовірності класів для кожної комірки. Він використовує рамки різних масштабів для захоплення об'єктів різних розмірів. Остаточні передбачення генеруються шляхом комбінування передбачень з кількох комірок сітки та рамок[1].

Застосовувати їх будемо на Jupyter Notebook – веб-додатку, який дозволяє розробляти код та текстові елементи різних форматів. У ньому зручно аналізувати дані, а також перевагою є те, що додаток працює локально, без доступу до Інтернету.



## РОЗДІЛ 3

### ЗАСТОВУВАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ВІЙСЬКОВИХ ЛІТАКІВ

#### 3.1 Підготовка даних

Для задачі класифікації будемо використовувати набір даних із сайту Kaggle[2], в якому міститься 6300 кольорових зображень, кожне з яких відповідає одному з 9 класів літаків, або відсутності літака на фото. В останньому класі містяться знімки фону, де зображена, наприклад, бетонна злітна смуга чи злітно-посадковий майданчик. Перелік класів та кількості зображень в них подано на рисунку 3.1

B-1	620
B-2	663
B-52	607
C-5	616
C-17	667
C-130	632
C-135	616
E-3	626
KC-10	638
BareLand	615

Рис. 3.1 – Відповідність кількості зображень кожному класу.

Кожне зображення подано у форматі RGB, тобто в ньому кожен піксель має 3 складові (червоний колір, зелений, синій), отже зображення подається як

набір трьох матриць, по одній на кожну складову. Елементами матриць є числове значення в діапазоні від 0 до 255 відповідно до відтінку кольору

Подамо також приклади зображень, які містить цей набір даних на рисунку 3.2.

**KC-10****E-3****B-2****C-135****C-130****B-1**

Рис. 3.2 – Приклади зображень з набору даних

Завантажені дані можуть мати різний розмір зображення, тому перед навчанням моделі їх треба привести до одного виду. В нашому випадку було вирішено приводити навчальні зображення до розміру 120x120 пікселів. Після цього потрібно нормалізувати вхідні дані, тобто привести значення пікселів від проміжку  $[0, 255]$  до  $[0, 1]$ , адже для машинного навчання такі числа значно кращі. Для цього ділимо значення кожного пікселя на 255.

Далі потрібно поділити дані на тестувальний та тренувальний набір для визначення похибок в подальшому і ефективного навчання. На тестування виділяємо 20% зображень, а тренувати модель будемо на 80%.

### 3.2 Побудова моделі

У зв'язку з тим, що вхідний набір даних для навчання має досить мало прикладів, потрібно збільшити їх кількість. Для цього будемо використовувати в нашій моделі попередній шар `data_augmentation` [3]. Цьому шару передаємо як вхідний параметр розміри підготовлених фотографій. В ньому додамо до існуючих даних зображення з цього ж набору, але випадково зближені на не більше ніж 7%, а також обернені на вибраний випадково кут в межах  $[0,180]$ . Опис алгоритму нарощування даних поданий на рисунку 3.3.

```
def build_model_MAC():
    # main model
    data_augmentation = tf.keras.Sequential(
        [
            tf.keras.layers.RandomFlip("horizontal",
                                       input_shape=(64,64,3)),
            tf.keras.layers.RandomZoom(0.07),
            tf.keras.layers.RandomRotation(180)
        ]
    )
```

Рис. 3.3 – Урізноманітнення навчальних даних

Опишемо основні шари нейронної мережі. Перший згортковий шар `Conv2D` призначений для багатоколірних зображень, йому передаємо параметри згортки. У цьому випадку ми хочемо, щоб шар вивчав 32 згортки. Він ініціалізує їх випадковим чином і в ході тренування визначить значення фільтра, які найкраще підходять для узгодження вхідних даних з їхніми класами. Параметр (3, 3) вказує на розмір фільтра.

Наступним йде шар об'єднання, де ми розділили зображення на частинки  $2 \times 2$  і вибрали максимальне значення на кожній з них. Після цього – ще два шари згортки і об'єднання.

Для класифікації використовуємо повністю зв'язаний шар, який застосовується у звичайних нейронних мережах. Він приймає на вхід масив із впорядкованих результатів попередніх шарів, а повертає масив із передбаченими ймовірностями приналежності зображення до певного класу.

В результаті отримаємо структуру моделі, подану на рисунку 3. 4

Layer (type)	Output Shape	Param #
sequential_16 (Sequential)	(None, 64, 64, 3)	0
conv2d_21 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_21 (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_22 (Conv2D)	(None, 29, 29, 64)	18496
max_pooling2d_22 (MaxPooling2D)	(None, 14, 14, 64)	0
conv2d_23 (Conv2D)	(None, 12, 12, 32)	18464
max_pooling2d_23 (MaxPooling2D)	(None, 6, 6, 32)	0
flatten_14 (Flatten)	(None, 1152)	0
flatten_15 (Flatten)	(None, 1152)	0
dense_14 (Dense)	(None, 450)	518850
dropout_7 (Dropout)	(None, 450)	0
dense_15 (Dense)	(None, 10)	4510

Рис. 3.4 – Остаточна схема нейронної мережі

Ми використовуємо оптимізатор Adam, обираємо відповідну категоріальну функцію втрат і оцінюємо точність результатів на кожному етапі процесу навчання.

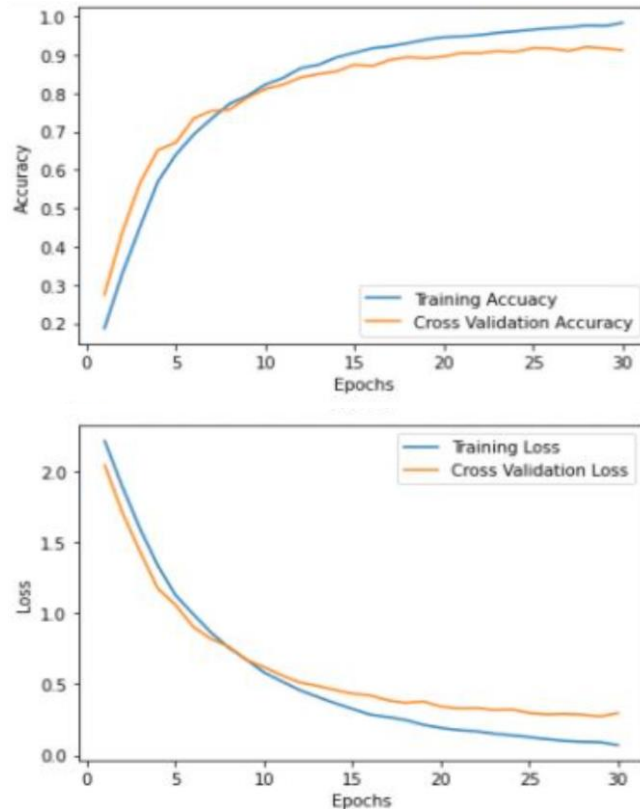


Рис. 3.5 – Точність і функції втрат на тренувальному та валідаційному наборі

Бачимо, що тренувальний набір показує точність близько 96%, а валідаційний – 91%, також функція втрат спадає як на тренувальному, так і на тестувальному наборі даних, при чому їх графіки досить близькі, отже перенавчання моделі не значне.

### 3.3 Модель для виявлення декількох об'єктів на фото

Для виявлення декількох об'єктів на фото перед їх класифікацією використаємо бібліотеку YOLOv5. Це глибока нейронна мережа, призначена саме для завдань виявлення об'єктів і знаходження їх рамок.

Імпотруємо бібліотеку, завантажуюємо модель і шукаємо з її допомогою всі літаки на фото.



Рис. 3.6 – Виявлені моделлю YOLOv5 літаки

Літаки на рисунку 3.6 були правильно класифіковані за допомогою попередньо описаної моделі як С-130



Рис. 3.7 – Виявлений літак за допомогою моделлю YOLOv5

Літак був класифікований як В-1 коректно.



Рис. 3.8 – Виявлені моделлю YOLOv5 літаки

Правильно класифіковані літаки класу C-130 показані на рисунку 3.8.

З рисунків бачимо, що модель виявлення об'єктів гірше справляється з літаками складної форми. Проте, модель класифікації досить добре розподіляє літаки на класи.

Однією з причин гіршого виявлення літаків B-1 може бути те, що модель YOLOv5 більше навчалась на літаках класичної форми. Способом її подальшого покращення може бути її доучання на додаткових прикладах.

## ВИСНОВКИ

Ми розглянули таку галузь машинного навчання, як згорткові нейронні мережі. Завдяки структурі нейронних мереж цього типу вони ефективно працюють для розпізнавання об'єктів, тому часто застосовуються для задач такого типу.

Під час розв'язування задачі було розроблено модель для класифікації літаків на зображеннях на 10 класів. Було вивчено способи задання зображень, а також методи їх обробки. Крім того ми дослідили залежність між характеристиками навчанням нейронної мережі і перенавчанням.

Після навчання модель досягла точності 91%, що є досить добрим результатом, особливо враховуючи невелику кількість зображень та їх роздільну здатність. У таких умовах буває важко розпізнати деталі, які дозволяють відрізнити об'єкти.

Для отримання кращих результатів можна збільшити навчальний набір літаків, або використовувати фотографії з більшим розміром. Проте, тоді потрібні будуть більші ресурси комп'ютера для навчання моделі.

Для можливості застосування моделі на прикладі реальних аерознімків було використано глибоку нейронну мережу YOLOv5. Вона краще обробляє літаки більш округлої форми, а ті, які мають складнішу форму, виявляє гірше. Для подальшого її покращення можна використати донавчання її на додаткових вхідних даних різноманітнішого плану.



## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ultralytics YOLOv8 Docs [Electronic resource]. – [Cited 10 June 2023]. – Available at:  
<https://docs.ultralytics.com/yolov5/>
2. Kaggle. Military Aircraft Recognition dataset [Electronic resource]. – [Cited 21 March 2023]. – Available at:  
<https://www.kaggle.com/datasets/khlaifiabile/military-aircraft-recognition-dataset>
3. TensorFlow Data augmentation [Electronic resource]. – [Cited 30 March 2023]. – Available at:  
[https://www.tensorflow.org/tutorials/images/data\\_augmentation](https://www.tensorflow.org/tutorials/images/data_augmentation)
4. TensorFlow | Python. [Electronic resource]. – [Cited 20 April 2023]. – Available at:  
[https://www.tensorflow.org/api\\_docs/python/tf](https://www.tensorflow.org/api_docs/python/tf)
5. Géron A. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow / Aurélien Géron, - 2019. - №2. P. 37 – 113
6. Chollet F. Deep Learning with Python/ F. Chollet. – 2018. P. 119 – 143
7. Moroney L. Praise for AI and Machine Learning for Coders/ Laurence Moroney – 2020. – №1. – С. 21-65
8. OpenCV [Electronic resource]. – [Cited 10 May 2023]. – Available at:  
<https://docs.opencv.org/4.x/index.html>
9. Overfit and underfit | TensorFlow Core. [Електронний ресурс]. – Режим доступу:  
[https://www.tensorflow.org/tutorials/keras/overfit\\_and\\_underfit#strategies\\_to\\_prevent\\_overfitting](https://www.tensorflow.org/tutorials/keras/overfit_and_underfit#strategies_to_prevent_overfitting)

10. Шефкін Б. В. ДОСЛІДЖЕННЯ ТА ВПРОВАДЖЕННЯ НЕЙРОННОЇ МЕРЕЖІ НА ОСНОВІ TensorFlow / Б. В. Шефкін, І. В. Красюк, В. О. Хоменчук, К. П. Сторчак, А. М. Тушич // ЗВ'ЯЗОК, № 6, 2020. С 20 – 24

## ДОДАТОК А. Програмна реалізація

Код програми можна знайти за посиланням:

<https://github.com/TetianaYaloveha/Diploma-work>