

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ
СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної
техніки Кафедра інформатики та програмної
інженерії

На правах рукопису»
УДК 4.93'12,004.93'14

«До захисту допущено»
Завідувач кафедри
_____ Едуард ЖАРІКОВ
«__» _____ 2022 р

Магістерська дисертація

на здобуття ступеня магістра

**за освітньо-професійною програмою «Інженерія програмного забезпечення
інформаційних систем»**

зі спеціальності 121 «Інженерія програмного забезпечення»

на тему: «Спосіб оперативної ідентифікації воєнної техніки

за допомогою машинного навчання»

Виконав :
студент II курсу, групи ІП-з11мп
Михайлов Данііл Євгенович

Керівник:
доцент кафедри ІПІ ФІОТ, к.т.н.

Новінський Валерій Петрович

Рецензент:
Доцент каф.ІСТ ФІОТ, к.т.н.
Попенко Володимир Дмитрович

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань

Студент _____

Київ – 2022 року

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря
Сікорського»**

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Рівень вищої освіти – другий (магістерський)

Спеціальність – 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма «Інженерія програмного забезпечення інформаційних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Едуард ЖАРІКОВ

«__» _____ 2022р.

**ЗАВДАННЯ
на магістерську дисертацію студенту
Михайлов Данііл Євгенович**

1. Тема дисертації «Спосіб оперативної ідентифікації воєнної техніки за допомогою машинного навчання», науковий керівник дисертації Новіньський Валерій Петрович доцент кафедри ІІІ ФІОТ, к.т.н., затверджені наказом по університету від «08» листопада 2022 р. No 4097-с
2. Термін подання студентом дисертації «12» грудня 2022 р.
3. Об'єкт дослідження – процес ідентифікації воєнної техніки за типами на знімках із супутника.
4. Предмет дослідження – методи машинного навчання для ідентифікації воєнної техніки.
5. Перелік завдань, які потрібно розробити – аналіз відомих підходів для ідентифікації об'єктів. Аналіз відомих методів машинного навчання. Розробка способу ідентифікації воєнної техніки на знімках із супутника. Аналіз та порівняння алгоритмів машинного навчання на основі представленого дата сету. Розробка стартап-проекту.
6. Орієнтовний перелік графічного (ілюстративного) матеріалу – 20

зображень.

7. Орієнтовний перелік публікацій – одна публікація

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ 1. Аналіз способів оперативної ідентифікації			
Розділ 2. Відомі підходи до ідентифікації			
Розділ 3. Опис розробки програмного забезпечення та використаних технологій			
Розділ 4. Розробка стартап-проекту			

9. Дата видачі завдання «08» листопада 2022 р. No 4097-с

Календарний план

/п	Назва етапів виконання магістерської дисертації	Термін виконання	Примітка
1	Затвердження теми	01.09.2022	
2	Вивчення та аналіз завдання	27.09.2022	
3	Пошук матеріалів по темі роботи	05.10.2022	
4	Розробка програмного продукту	20.10.2022	
5	Оформлення пояснювальної записки	15.11.2022	
6	Подання дисертації на попередній захист	22.11.2022	
7	Подання дисертації на захист	06.12.2022	

Студент

Данііл МИХАЙЛОВ

Науковий керівник

Валерій НОВІНСЬКИЙ

РЕФЕРАТ

Виконану на тему: Спосіб оперативної ідентифікації воєнної техніки за допомогою машинного навчання

студентом: Михайловим Даніілом Євгеновичем

Структура і обсяг роботи:

Робота складається із вступу та чотирьох розділів. Загальний обсяг роботи: аркушів основного тексту, 20 ілюстрацій, 23 таблиці. При підготовці використовувалася література з різних джерел.

Актуальність роботи.

Як показує сьогодення, питання ідентифікації воєнної техніки на знімках із супутника стає дедалі актуальніше. Якщо з'явиться можливість швидко ідентифікувати техніку на знімках із супутника це пришвидшить роботу наших воєнних експертів і нашу Перемогу.

Супутникові зображення стали невід'ємною частиною в сучасних конфліктах. Тому я хотів дослідити можливість використання моделей глибокого навчання для точної ідентифікації воєнної техніки та її класифікації, щоб полегшити роботу наших військових та збільшити їх швидкість реагування та покращити розвіддані.

Мета роботи і завдання дослідження: Метою магістерської роботи є розробка ідентифікації воєнної техніки та її типів за знімком із супутника.

Для досягнення мети дослідження поставлено і вирішено такі завдання:

- огляд існуючих рішень у біометричній ідентифікації та комп'ютерного зору;
- вибір та розробка способу ідентифікації: визначення вимог, знаходження даних джерел, опис програмних засобів, функцій та використаних моделей;

- розробка та тестування системи: навчання нейронних мереж, тестування системи, збір аналітичних даних, аналітика на основі тестування.

Об’єкт дослідження: Процес ідентифікації воєнної техніки, її класифікація на основі знімків з супутника.

Предмет дослідження: Методи ідентифікації воєнної техніки за даними із дата сету з використанням технології згорткових нейронних мереж, багатошарового перцептрону, та модифікованої згорткової мережі.

Методи дослідження: Для виконання поставлених завдань було використано: технології згорткових нейронних мереж, багатошарового перцептору з машинного навчання. Дані методи були використані для аналізу та порівняння результатів прогнозування, для подальшого створення модифікованого алгоритму прогнозування.

Наукова новизна. Науковою новизною є отриманий метод ідентифікації воєнної техніки на основі датасету. Аналіз та виявлення найбільш оптимального методу для класифікації воєнної техніки.

Практичне значення одержаних результатів. Результати досліджень можуть бути використані для подальшої розробки системи ідентифікації об’єктів чи ідентифікації у реальному часі, та використовуватися під час воєнних дій.

Ключові слова: аналіз, машинне навчання, штучний інтелект, модель, класифікація, датасет.

ABSTRACT

Structure and scope of master's thesis:

The work consists of an introduction and four chapters. Total volume of work: sheets of the main text, 20 illustrations, 23 tables. Literature from various sources was used during the preparation.

Relevance of work.

As the present shows, the question of identifying military equipment on satellite images is becoming more and more urgent. If it becomes possible to quickly identify equipment on satellite images, it will speed up the work of our military experts and our Victory.

Satellite images have become an integral part of modern conflicts. Therefore, I wanted to explore the possibility of using deep learning models to accurately identify military equipment and classify it in order to facilitate the work of our military and increase its response speed and improve intelligence.

Purpose of work.

The purpose of the master's thesis is to develop the identification of military equipment and its types based on a satellite image.

To achieve the goal of the research, the following tasks were set and solved:

- overview of existing solutions in biometric identification and computer vision;
- selection and development of an identification method: definition of requirements, finding a dataset, description of software tools, functions and used models;
- system development and testing: neural network training, system testing, analytical data collection, test-based analytics.

Object of research: The process of identification of military equipment, its classification based on satellite images.

Subject of research: Methods of identification of military equipment based on data from the dataset using the technology of convolutional neural networks, multilayer perceptron, and modified convolutional network.

Research methods: To perform the assigned tasks, the following technologies were used: convolutional neural networks, multilayer perceptron from machine learning. These methods were used to analyze and compare forecasting results, to further create a modified forecasting algorithm.

Scientific novelty: The obtained method of identification of military equipment based on a dataset is a scientific novelty. Analysis and identification of the most optimal method for the classification of military equipment.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	10
ВСТУП	11
РОЗДІЛ 1. АНАЛІЗ СПОСОБІВ ОПЕРАТИВНОЇ ІДЕНТИФІКАЦІЇ	13
1.1 Аналіз алгоритмів ідентифікації на основі комп'ютерного зору	13
1.2 Аналіз підходів використання шаблонів та нейронних мереж	17
1.3 Переваги та недоліки існуючих згорткових нейронних мереж	19
1.4 Аналіз новизни застосування ідентифікації техніки	22
ВИСНОВКИ ДО РОЗДІЛУ 1	24
РОЗДІЛ 2. ВІДОМІ ПІДХОДИ ДО ІДЕНТИФІКАЦІЇ	25
2.1 Сучасні способи ідентифікації	25
2.2 Теорія структури нейронних мереж для обробки зображень	29
2.2.1 Архітектура багатошарових нейронних мереж	29
2.2.2 Нейронні мережі високого порядку та моментні НР	32
2.2.3 Радіально-базисні нейронні мережі	33
2.3 Теорія способів машинного навчання	35
2.3.1 Глибинне структуроване навчання	35
2.3.2 Кероване навчання	37
2.3.3 Напівкероване навчання	38
2.3.4 Неконтрольоване навчання	40
2.3.5 Навчання з підкріпленням	41
2.3.6 Регресійний аналіз	42
2.3.7 Навчання на прикладах	44
ВИСНОВКИ ДО РОЗДІЛУ 2	45
РОЗДІЛ 3. ОПИС РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ВИКОРИСТАНИХ ТЕХНОЛОГІЙ	46
3.1 Спосіб ідентифікації техніки та його застосування	46
3.1.2 Multilayer perceptron	48
3.1.3 ЗГОРТКОВА НЕЙРОННА МЕРЕЖА	51
3.2 Засоби розробки	52
3.3 ДАТАСЕТ	52
3.3.1 Метрики оцінювання	55
Результати практичних іспитів	55

ВИСНОВКИ ДО РОЗДІЛУ 3	64
РОЗДІЛ 4. РОЗРОБКА СТАРТАП-ПРОЕКТУ	64
4.1 Опис ідеї проекту	65
4.3 Аналіз ринкових можливостей запуску стартап-проекту	69
4.3 Аналіз інноваційності проекту	80
ВИСНОВКИ ДО РОЗДІЛУ 4	89
ВИСНОВКИ	90
ПЕРЕЛІК ПОСИЛАНЬ	91
ДОДАТКИ	94

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ECFA – Evolution-Constructed Features Algorithm.

SIFT – Scale Invariant Feature Transform.

SURF – Speeded-Up Robust Features.

БНМ – багатошарова нейронна мережа.

НМ – нейронна мережа.

НМВП – нейронна мережа високого порядку.

РБНМ – радіально-базисні нейронні мережі.

МПВ – марковський процес вирішення.

CNN – згорткова нейронна мережа

ВСТУП

Експоненційна швидкість розвитку технологічного прогресу вражає своїм діапазоном та впливом на якість життя як усього людства, так і кожної окремої людини. Виходячи з аналізу історичної хронології розвитку науки, найвпливовіші розробки виникали спочатку з військовими цілями, і вже потім ставали двигунами прогресу в післявоєнний час. Таким чином, будь-які технології, що створюються в кризові проміжки історії для розв'язання цієї проблеми, мають високий потенціал стати затребуваними та практичними у мирний час. Однією з таких ідей є спосіб оперативної ідентифікації військової техніки за допомогою нейронних мереж.

Мабуть, найпопулярнішим завданням нейромереж є розпізнавання візуальних образів. Сьогодні створюються мережі, в яких машини здатні успішно розпізнавати символи на папері та банківських картках, підписи на офіційних документах, детектувати об'єкти тощо.

Оскільки супутникові зображення стали дуже важливими в сучасних конфліктах, я хотів дослідити можливість використання моделі глибокого навчання для ідентифікації та точної класифікації різних типів військових транспортних засобів, які можуть бути знайдені на полі бою. Дедалі більше використання супутникових зображень для відстеження конфліктів призвело до відповідного збільшення величезного обсягу створюваних зображень, і завдяки цьому рішення, розроблене для автоматичної ідентифікації різних транспортних засобів, може допомогти спостерігачам переглядати тисячі зображень і швидко аналізувати ті, що містять предмети інтересу.

Метою цієї роботи є пошук, аналіз та розробка способу ідентифікації, що має працювати з зображеннями воєнної техніки з супутника та знаходити її з високою точністю.

Завданням цієї роботи буде проаналізувати існуючі алгоритми нейронних мереж, та модифікувати найбільш оптимальний для покращення результату в поставленій задачі.

Це можна зробити за аналогією зі схожими системами, що вже впроваджені у життя – це системи контролювання доступу, охоронні та фінансові системи, тощо. Влучним прикладом, що найбільш схожий на тематику цієї роботи, може стати система відеофіксації порушень ПДД, де завданням камери є пошук, ідентифікація транспорту та фіксація правопорушення.

Об’єкт дослідження: Процес ідентифікації воєнної техніки, її класифікація на основі знімків з супутника.

Предмет дослідження: Методи ідентифікації воєнної техніки за даними із даних сету з використанням технології згорткових нейронних мереж, багатошарового перцептрону, та модифікованої згорткової мережі.

Мета досліджень: Підвищення оперативності та достовірності ідентифікації воєнної техніки за цифровим супутниковим знімкам.

РОЗДІЛ 1. АНАЛІЗ СПОСОБІВ ОПЕРАТИВНОЇ ІДЕНТИФІКАЦІЇ

1.1 Аналіз алгоритмів ідентифікації на основі комп'ютерного зору

Завдання розпізнавання образів є основним у більшості інтелектуальних систем, серед яких можна виділити декілька прикладів.

Оскільки існує багато завдань, пов'язаних з розпізнаванням об'єктів (наприклад, розпізнавання об'єктів і людей на фотографіях, розпізнавання об'єктів у відеопотоку, відстеження траєкторій і т.д.) слід визначити тип даних, що входять до алгоритмів розпізнавання. Під «відеопотоком» справжньої роботи буде матися на увазі відео, отримане з автомобільного відеореєстратора в Full HD якості.

Аналізуючи тенденції розвитку застосування алгоритмів розпізнавання, можна зробити висновок, що пошук рішень таких завдань, як розпізнавання дорожніх знаків та обстановки на дорозі є найближчою до цієї роботи темою застосування ідентифікації об'єктів в даний час. Дослідженню алгоритмів для побудови систем, що вирішують ці завдання, присвячено декілька наступних, розглянутих нами, статей.

У багатьох статтях виконано огляд існуючих алгоритмів, застосовуваних у розпізнаванні, і навіть було розглянуто низку систем, застосовуваних розпізнавання суто дорожніх знаків. Ці знання нам знадобляться для спеціалізації системи під ідентифікацію суто воєнної техніки. Крім цього, у одній з статей було знайдено аналіз та порівняння характеристик відомих на даний момент комерційних та експериментальних систем розпізнавання дорожніх знаків.

Стислий розгляд літературного напрацювання попередніх колег з цієї теми можна звести до наступних ідей.

Комп'ютерний зір – одна з сучасних комп'ютерних технологій, що розвиваються. Ця область має у своїй суті пошук різних алгоритмів для розпізнавання фігур, об'єктів, а також вибір найбільш оптимальних алгоритмів, які можна застосувати для тих чи інших завдань. Наприклад, можна виділити

такі сфери комп'ютерного зору, як розпізнавання осіб та розпізнавання дорожніх знаків на зображеннях.

Було розглянуто та проаналізовано ряд експериментів над алгоритмами, що використовують у своїй основі технологію комп'ютерного зору.

Value канал з обробкою Gaussian Blur 9×9 за роботою алгоритмів Canny–Hough показав дуже багато хибних спрацювань. Такі об'єкти, як дерева, дають дуже багато перешкод. Знайдено 60 % круглих символів. Межі прямокутних знаків нечіткі, складаються із серії прямих ліній.

За збільшенням контрастності зображення алгоритмами програм кількість хибних спрацювань візуально зменшується, на відміну від попереднього експерименту, але залишається досить високою у порівнянні з результатами 1-го експерименту.

За збільшенням пастеризації з'являються нові артефакти (зменшення кількості кольорів створює нові об'єкти).

На даний момент є ряд систем, які реалізують різні підходи в розпізнаванні об'єктів (наприклад, конкретних дорожніх знаків). У більшості випадків використовуються нейромережі, які класифікують даний об'єкт з певною ймовірністю. У той же час дані системи вимагають передобробку вхідних даних і виділення області, що цікавить нас, із загального масиву даних (наприклад, виявлення регіону інтересу в кадрі).

Можна зустріти нові алгоритми, такі як ECFA, які показують більш перспективні результати, ніж існуючі алгоритми та системи. Однак через новизну даних алгоритмів, важко знайти їх детальний опис і відкриту реалізацію ідеї.

Кожен із розглянутих алгоритмів (SIFT, SURF та ін) мають ряд переваг та недоліків, що накладають відбиток на їх сферу застосування. Алгоритми SIFT, SURF найбільше підходять для складання зображень при побудові панорамних знімків, моделювання 3D сцен, розпізнавання людських дій, визначення положення роботів на основі порівняння з 3D картою, що не є актуальним для

нашої роботи. Тим не менш, вони можуть бути використані у випадках порівняння ключових точок вихідного зображення з ключовими точками з існуючої БД (наприклад, БД дорожніх знаків у статтях, або БД воєнної техніки у нашій роботі).

Одна з статей рекомендувала алгоритми Canny-Hough, як найбільш ефективні. На основі отриманих даних були зроблені такі висновки: Алгоритми Canny-Hough вимагають вхідне зображення/відео дуже гарної якості, зняте за хороших погодних умов днем, або з добре підбраною системою фільтрації та шумоподавлення. У разі проведених експериментів незначна перешкода була також алгоритмами стиснення відео. Таким чином, робота з супутниковими зображеннями не підходить цьому алгоритму, але ми можемо дещо взяти звідти.

Таким чином усі існуючі алгоритми ідентифікації об'єктів, що будуть використовуватися для подальшої ідентифікації воєнної техніки, потребують значного покращення роботи алгоритмів детекції. Є вірогідність, що у процесі роботи нам знадобиться пастеризація на основі класів кольорів та покращення алгоритмів попередньої обробки зображення для покращення якості роботи алгоритмів ідентифікації, чого немає у більшості існуючих програм, особливо у алгоритмів Canny.

Крім цього, за попереднім аналізом статей та практичного використання комп'ютерного зору був зроблений висновок не використовувати цю технологію у подальшій розробці програми через її недостатньо високу точність задля виконання завдань цієї роботи.

Цей алгоритм має критичні наявні недоліки, такі як відсутність підтримки чорних і білих об'єктів, вплив об'єктів того ж кольору, що й об'єкт. Оскільки граничні значення задані статично, на даний алгоритм можуть впливати зовнішні фактори: освітлення, погодні умови та якість фото зі супутника.

Існує низка та інших алгоритмів. Одним з них можна назвати алгоритм багатомасштабних згорткових нейронних мереж. Вони мають високий ступінь ефективності, є вузькоспеціалізованими але вимагають навчання

1.2 Аналіз підходів використання шаблонів та нейронних мереж

Завдання розпізнавання техніки на зображенні є одним із стандартних завдань комп'ютерного зору. Програмне забезпечення, що постачається разом із системами відеоспостереження, в більшості випадків, пропонує функціонал розпізнавання автомобільних номерів, тобто аналогічні системи вже впроваджені у життя. У той же час, існує безліч інших підходів до вирішення цієї задачі.

Найбільш поширеними є підходи, засновані на використанні шаблонів. Такі підходи пропонують передбачуваний і невеликий час виконання та низький відсоток помилок. Проте, ці підходи виявляється набагато менш ефективними, коли постає необхідність розпізнати різну воєнну техніку, яка може перебувати в непередбачуваному місці, бути не дуже гарної якості і бути гарно замаскована.

Одним із способів збільшити ефективність і якість розпізнавання такої техніки є використання нейронних мереж.

Знайомлячись із статтями на цю тему, можна переконатися у тому, що оптимальним методом вирішення цієї задачі є застосування шаблонів.

І, дійсно, багато систем розпізнавання об'єктів (наприклад автомобільних номерів) використовують методи на застосуванні шаблонів. Суть таких методів полягає в тому, що алгоритм має інформацію про те, як виглядає необхідний об'єкт, яким у нього може бути фон, як виглядають і на яких позиціях можуть бути певні символи (диференціюються літери і цифри) , враховує шрифт, який використовується для їх накреслення, відразу розглядає можливе місце виявлення об'єкту, а не все зображення. Все це дозволяє досягти високої якості розпізнавання стандартизованих номерних знаків. Продуктивність таких методів більш ніж задовільна. Зазвичай, впевненість авторів, що захищають методи, засновані на шаблонах, впливає з того факту, що вони розглядають застосування своїх методів у межах однієї спеціалізованої сфери.

Однак, який метод застосовувати у випадку, якщо необхідно розпізнавати об'єкти серед скупчення схожих зображень, що задовольняють різні шаблони? Звісно є можливість використовувати безліч шаблонів, на шкоду ефективності. Або ускладнити один шаблон, зробивши його більш універсальним, на шкоду якості. Але який метод використовувати, якщо нам необхідно розпізнавати різні види воєнної техніки поза форматом, що знаходяться в несподіваному місці, різним чином відображені на знімку супутника, розміщені на різному тлі.

Можливою відповіддю на всі ці питання є використання штучних нейронних мереж. У машинному навчанні та в когнітивістиці, під штучними нейронними мережами розуміють сімейство моделей, натхненних біологічними нейронними мережами (центральною нервовою системою тварин, зокрема – мозком), і використовуються для того, щоб оцінити або наблизити функції, які можуть залежати від величезної кількості вхідних даних і зазвичай невідомі.

Як і інші методи машинного навчання – системи, що навчаються з даних – нейронні мережі використовуються для вирішення безлічі завдань, які важко вирішити з використанням стандартного логічного програмування, включаючи завдання комп'ютерного зору.

Існує кілька видів штучних нейронних мереж. Відрізняють feed-forward (від англ. feed - годувати і forward - вперед) і рекурентні нейронні мережі.

По-перше – зв'язки між нейронами не можуть формувати цикл, по друге – формують, з метою надання зворотного зв'язку більш верхнім рівням. Також, нейронні мережі класифікують виходячи з їхньої архітектури – багатошаровий перцептрон (багатошарові нейронні мережі), глибокі (складний багатошарові нейронні мережі, що виробляють безліч нелінійних перетворень), згорткові (що містять згорткові шари) і інші

1.3 Переваги та недоліки існуючих згорткових нейронних мереж

У машинному навчанні, згорткові нейронні мережі є видом feed-forward штучних нейронних мереж, в яких індивідуальні нейрони закладені таким чином, що вони відповідають регіонам, що перекриваються, у полі зору.

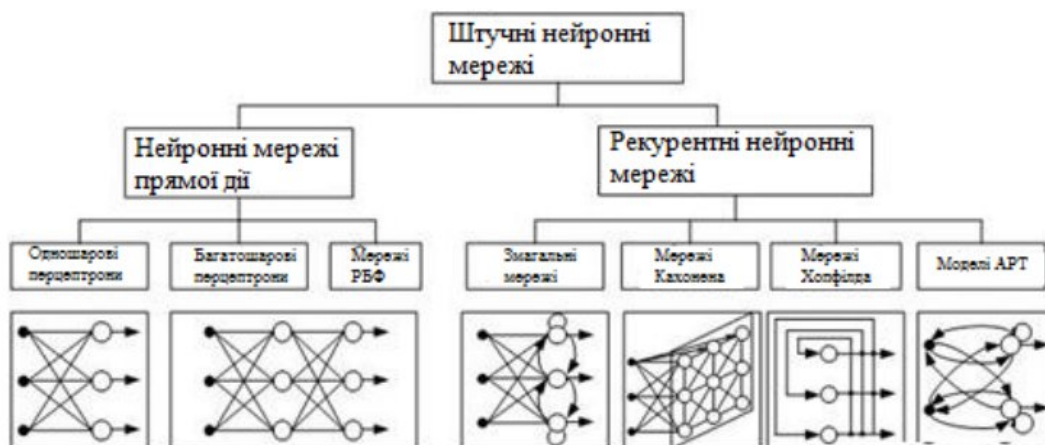


Рисунок 1.1 – Класифікація нейронних мереж за архітектурою

Згорткові нейронні мережі були натхненні біологічними процесами і є варіацією багатошарового персептрону, спроектованого для використання мінімального об'єму передобробки. Вони знаходять широке застосування в обробці відео та зображень, що рекомендують системах та обробці природної мови.

Якщо такі мережі використовуються для розпізнавання зображень, вони складаються з безлічі шарів з невеликих скупчень нейронів, які беруть невеликі частини вхідного зображення. Називаються ці скупчення - рецептивними полями. Результат роботи даних скупчень потім розміщується так, що вони перетинаються для того, щоб набути кращого представлення оригінального зображення; це повторюється для кожного наступного шару.

Проте, із застосуванням нейронних мереж пов'язані кілька складнощів – складність навчання мереж та низька продуктивність. Для того, щоб вирішити

другу проблему, згорткові мережі можуть містити локальні і глобальні верстви субдискретизації (subsampling, pooling шари), які комбінують виведення кластерів нейронів (Рисунок 1.2), тим самим знижуючи кількість (і деталізацію) інформації.

Якщо правильно підібрати параметри таких шарів, можна істотно підвищити продуктивність мережі, не втративши, при цьому, у якості.

У розглянутих у ході літературної роботи статтях було спроектовано кілька згорткових нейронних мереж для ідентифікації визначених об'єктів з використанням фреймворку для побудови нейронних мереж Caffe.

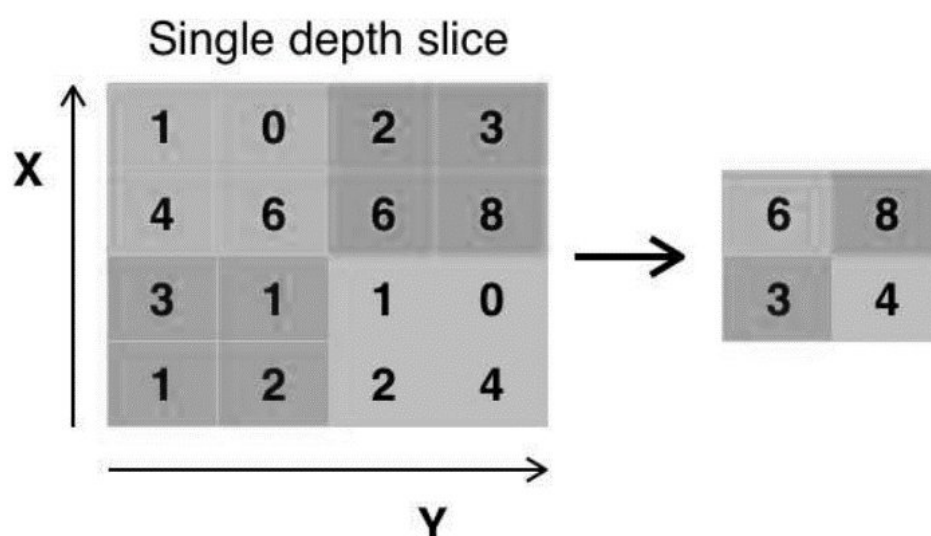


Рисунок 1.2 – Приклад роботи шару субдискретизації

Згорткові нейронні мережі також як і каскадні класифікатори з ознаками Хаара показують досить високий рівень точності та повноти при пошуку об'єктів на зображенні. Нейронні мережі добре справляються з різними варіаціями нахилу об'єкта, поки каскадні класифікатори нестійкі до нахилених об'єктів на зображенні. Однак, щоб забезпечити високу стійкість мережі в таких ситуаціях, необхідно урізноманітнити навчальну вибірку зображеннями з подібними перетвореннями.

У той час як каскадні класифікатори з ознаками Хаара мають високу стійкість до зміни масштабу об'єктів на зображеннях, згорткова нейронна мережа

в деяких випадках не може розпізнати об'єкт при зміні його масштабу. Для навчання нейронної мережі необхідно підготувати велику та якісну навчальну вибірку.

В результаті проведених досліджень, у ряді тестових умов (були використані різні набори завдань у різних статтях) вдалося досягти якості розпізнавання об'єктів, що перевершує шаблонний метод. Недоліком може стати той факт, що нейронна мережа перевищує швидкість шаблонів лише на незначний проміжок, бо багато часу витрачається на навчання алгоритмів. З іншого боку це ідеальне рішення для використання у даній роботі для ідентифікації воєнної техніки з високою точністю, але це потребує багато часу на навчання нейронної мережі.

Завданням цієї роботи буде запропонувати нові способи оптимізації нейронних мереж для прискорення швидкості їх навчання зі збереженням якості.

1.4 Аналіз новизни застосування ідентифікації техніки

Натренована на знаходження конкретної техніки та обладнання нейронна мережа є досить спеціалізованим інструментом, але діапазон застосування таких алгоритмів дуже великий. З перегляду літературних джерел та практичного застосування алгоритмів ідентифікації об'єктів ми дійшли висновку, що такі програми мають дуже багато галузей призначення, але дуже рідко зустрічаються у воєнних розробках.

Така інформація найчастіше буває секретною, тому поглиблених матеріалів з таких розробок немає у вільному доступі зовсім. Тому є сенс припустити, що доопрацьована система детекції воєнної техніки на знімках супутників є досить новою ідеєю, що відрізняється завданням від алгоритмів ідентифікації дорожніх знаків, автомобільних номерів та людей, але має схожу конструкцію.

Найбільш актуальним та інтуїтивно зрозумілим завданням для такої програми є збір та обробка будь-яких матеріалів розвідки. Це може бути

- попередній аналіз фото- та відео-матеріалів алгоритмами з метою розвідки до початку безпосередньо бойових дій;
- аналіз фото- та відео-матеріалів алгоритмами з метою розвідки у момент бойових дій для коригування вогню та аналізу ситуації на фронті і на територіях, що прилягають до театру воєнних дій;
- алгоритмізований пошук диверсантів та контроль дисципліни у власних військових частинах;
- нейронні мережі можна використовувати у безпілотних літальних апаратах у якості регулювання системи прицілювання;
- використання у винищувачах та гелікоптерах впроваджених алгоритмів задля підвищення якості систем прицілювання та орієнтації на місцевості;
- охорона та спостереження великих територій, військових частин, сільського господарства;

- аналіз алгоритмами нейронних мереж дорожнього трафіку з метою адаптування роботи світлофорів для комфортного регулювання заторів у визначенні часи перенавантаження.

ВИСНОВКИ ДО РОЗДІЛУ 1

Результатами проведених аналізів у першому розділі є огляд літературних джерел, що пов'язані з темою роботи, та визначення невирішених питань, а саме конкретизація даного дослідження у розв'язанні проблеми ідентифікації воєнної техніки. Були сформульовані основні недоліки існуючих досягнень у розробці алгоритмів ідентифікації, та розроблені напрямки їх подальшого усунення та вирішення у роботі над основним питанням дисертації.

Другий розділ присвячений розгляду, аналізу та формулюванню вимог до основного методу, який планується до використання у роботі. Для цього було опрацьовано матеріал з видів комп'ютерного зору, розглянути існуючі алгоритми роботи з методами шаблонів, проаналізовані переваги та недоліки нейронних мереж та на їх основі був синтезований новий, актуальний для теми роботи, варіант розробки та покращення програми ідентифікації воєнної техніки на зображеннях знімків супутників. Таким чином у цьому розділі ми намагалися вирішити, на основі яких технологій буде реалізоване навчання і адаптація нейронної мережі програмного забезпечення.

Крім того, було розглянуто деякі з можливих сучасних найбільш розповсюджених варіантів навчання – глибинне структуроване навчання, кероване навчання, напів кероване навчання, неконтрольоване навчання, регресійний аналіз, навчання на прикладах. Кожен з них має свої переваги та недолік у різних задачах, для використання був обраний вправний синтез для найбільшої оптимізації зусиль. Результати цієї аналітики наведені найбільш близькими до теми роботи прикладами, що спеціалізовані на роботі з воєнними об'єктами, хоча з іншої сторони, така програма може стати у нагоді при роботі і з цивільною інфраструктурою.

РОЗДІЛ 2. ВІДОМІ ПІДХОДИ ДО ІДЕНТИФІКАЦІЇ

2.1 Сучасні способи ідентифікації

З загального визначення ідентифікації можемо виділити те, що основним у цьому процесі є встановлення тотожності невідомого об'єкта відомому на підставі збігу наперед заданих ознак.

Для інформаційних систем використовують визначення – ідентифікатор, що є більш розширеним поняттям ознаки, що однозначно ідентифікує цього суб'єкта.

Таким чином, кожен об'єкт, явище, властивість, що потрібно підвести під ідентифікацію, має певний набір ознак, що виділяють його з багатьох інших. Відмінність одного об'єкта від іншого складає основі певних ознак, властивих цим об'єктам. Ця робота бере за необхідність ідентифікацію конкретного об'єкту чи групи подібних об'єктів, а саме воєнну техніку та обладнання.

Серед найбільш актуальних методів ідентифікації об'єктів можна виділити такі:

- унікальних найменувань;
- цифрових номерів;
- умовних позначень;
- класифікаційний;
- посилальний;
- описовий;
- описово-посилальний.

Найбільш близький до нашої задачі є описовий метод ідентифікації. Він використовується у випадках, коли необхідно ідентифікувати конкретний об'єкт шляхом опису його властивостей, параметрів, показників. Однією з основних переваг цього методу ідентифікації є можливість здійснення порівняльного

аналізу однорідних (родинних) об'єктів шляхом порівняння характеристик, що увійшли до їхньої ідентифікації.

Крім цього також можна розглянути описово-посилальний метод ідентифікації. Він, на відміну від описового, використовує лише частину основних характеристик об'єкта у поєднанні з посиланням на документ, де вміщені всі його характеристики. Найбільш широко цей метод використовується при створенні банків даних про різні об'єкти, а також про різні інформаційні видання, такі як каталоги, покажчики, кадастри тощо. Він дозволяє значно скоротити обсяг інформації, необхідний для ідентифікації об'єктів, що має суттєве значення для економії комп'ютерної пам'яті та скорочення обсягів роботи.

Ці логічні методи роботи з даними для ідентифікації, реалізовані у тілі згорткових нейронних мереж, повинні мати високу ефективність.

Повнозв'язкова нейронна мережа (яку ще називають звичайною) у цьому випадку не дуже підходить до нашої задачі. У ній кожен вузол (крім вхідного та вихідного) виступає як входом, так і виходом, утворюючи прихований шар нейронів, і кожен нейрон наступного шару з'єднаний з усіма нейронами попереднього. Входи подаються з терезами, які в процесі навчання налаштовуються і не змінюються надалі. При цьому кожен нейрон має поріг активації, після проходження якого він приймає одне з двох можливих значень: -1 або 1, або 0 або 1.

У той самий час, загорткова НМ має спеціальну архітектуру, яка дозволяє їй максимально ефективно розпізнавати образи. Сама ідея НМ ґрунтується на чергуванні загорткових та субдискретизуючих шарів (pooling), а структура є односпрямованою. НМ отримала свою назву від операції згортки, яка передбачає, що кожен фрагмент зображення буде помножити на ядро згортки поелементно, при цьому отриманий результат повинен підсумовуватися і записатися в схожу позицію вихідного зображення.

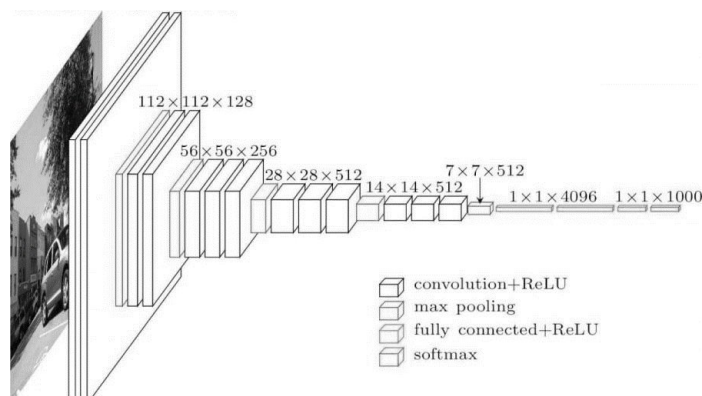


Рисунок 2.1 – Візуальна модель загорткової нейронної мережі

Ідентифікувати (розпізнати) об'єкт як воєнну техніку – означає, на підставі збігу вказаних ознак, підібрати відповідну відому інформацію про транспорт/обладнання. Тобто система, що виконує цей процес автоматично, має приблизний набір завдань:

- розбиття простору ознак на області, що відповідають класам (класифікація, розпізнавання, кластеризація);
- вилучення ключових характеристик, стиснення та реконструкція образів;
- апроксимація функції багатьох змінних з будь-якою заданою точністю;
- прогнозування часових рядів;
- асоціативна пам'ять;
- вирішення оптимізаційно-комбінаторних завдань;
- топологічно впорядковане перетворення простору;
- розпізнавання з урахуванням топології простору.

Більшість із цих завдань прямо чи опосередковано пов'язані з методами ідентифікації (розпізнаванням зображень). Таким чином, точний спосіб визначатиметься на основі ознак, за якими буде відбуватися ідентифікація. Найбільший інтерес до себе притягує поділ простору ознак на області та вилучення ключових ознак.

Для вирішення цих завдань призначені багатошарові нейронні мережі, нейронні мережі високого порядку та радіально-базисні нейронні мережі. Оскільки такі мережі оперують у вихідному просторі зображень (ознак), то для них вимога перед обробки зображення є критичною. Це приведення зображення до стандартного вигляду (положення, масштаб, орієнтація, вирівнювання яскравості), зниження розмірності даних, вибір ключових характеристик.

Наступним наслідком оперування у вихідному просторі є неможливість обліку спотворення зображення (наприклад, при зміні ракурсу зображення техніки), і тому тренувальна вибірка повинна містити репрезентативний набір прикладів, що є набором зображень об'єктів у тому діапазоні ракурсів та умов освітлення, в яких планується застосування системи розпізнавання.

2.2 Теорія структури нейронних мереж для обробки зображень

2.2.1 Архітектура багат шарових нейронних мереж

Архітектура багат шарової нейронної мережі (БНМ, багат шаровий персептрон) складається з послідовно з'єднаних шарів, де нейрон кожного шару своїми входами пов'язаний з усіма нейронами попереднього шару, а виходами – наступного. Активаци́йний функціями для таких нейронів служать різновиди лінійних, порогових та сигмоїдних функцій.

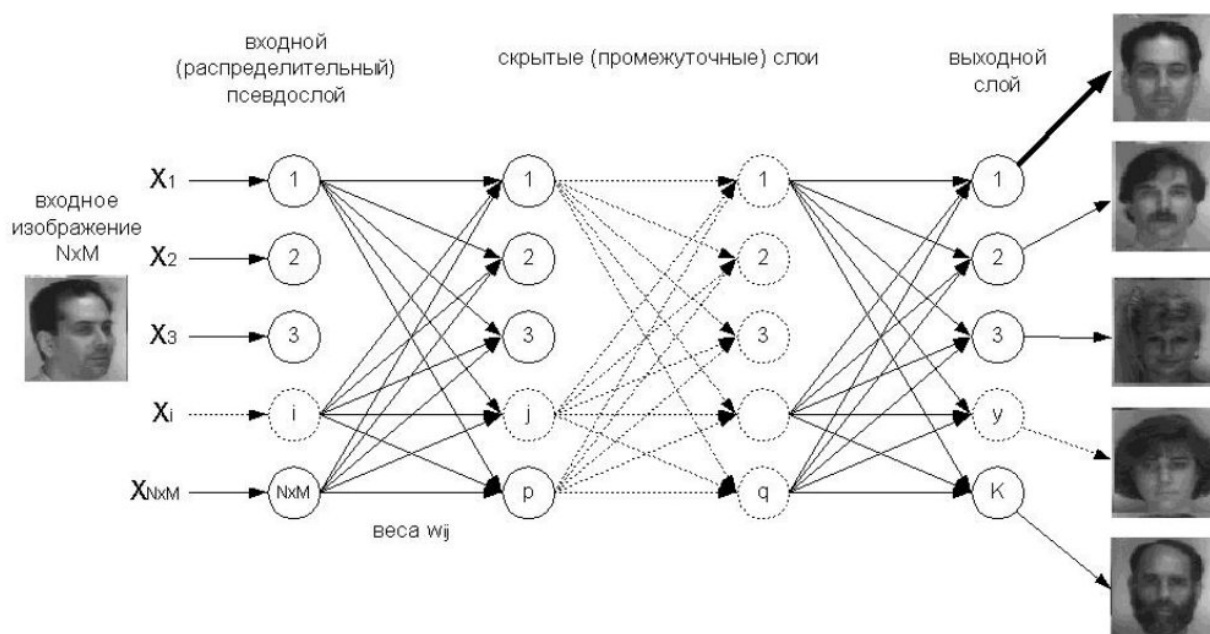


Рисунок 2.2 – Архітектура багат шарової нейронної мережі та її застосування для розпізнавання зображень.

НМ з одним вирішальним шаром здатна формувати лінійні розділяючі поверхні, що сильно звужує коло розв'язування задач, зокрема, така мережа не може вирішити задачу типу «що виключає або». НМ з нелінійною функцією активації та двома вирішальними шарами дозволяє формувати будь-які опуклі області у просторі рішень, а з трьома вирішальними шарами – області будь-якої складності, у тому числі й не випуклої форми.

При цьому БНМ не втрачає своєї узагальнюючої здатності. За допомогою двошарової НМ можна з будь-якою точністю апроксимувати будь-яку багатовимірну функцію на відрізку. Навчаються БНМ за допомогою алгоритму зворотного поширення помилки, що є різновидом градієнтного спуску в просторі терезів з метою мінімізації сумарної помилки мережі:

$$\Delta W = -\alpha \frac{dE}{dW}, \quad E = \frac{1}{2} \sum_j (y_i - t_j)^2$$

де t_j – еталонне значення виходів мережі. При цьому помилки (точніше, величини корекції ваги) поширюються у зворотному напрямку від виходів до виходів, крізь ваги, що з'єднують нейрони. Алгоритм зворотного поширення помилки є NP-важким, тому час навчання мережі збільшується експоненційно зі зростанням розмірності даних.

Інший напрямок розвитку архітектури БНМ – це нейронні дерева. У цій архітектурі вузлами вирішального дерева є нейронні мережі. У міру просування від кореня дерева мережі-вузли уточнюють розв'язання задачі. Точність розпізнавання зображень таких алгоритмів була вищою, але їх загальні характеристики не перевірялись.

Не дивлячись на те, що в більшості функцій БНМ ідеально виконують завдання цієї роботи, вони все одно мають кілька проблем:

- 1) Проблема локального мінімуму;

Як і для всіх градієнтних методів, проблема локального мінімуму полягає в тому, що при ітераційному спуску може настати момент, коли рішення входить у локальний мінімум, з якого внаслідок малої величини кроку не може вибратися. І такий локальний мінімум не завжди забезпечує прийнятне рішення. Вихід полягає у застосуванні стохастичних методів.

- 2) Вибір архітектури мережі;

З цим також пов'язана проблема перенавчання, яка полягає в тому, що мережа з надмірною кількістю елементів втрачає узагальнюючу здатність і добре працює тільки на тренувальній вибірці. В даний час розроблено різні апріорні оцінки вибору архітектури, методи проріджування навчених мереж, методи «зростаючих» мереж.

3) Вибір кроку (швидкості) навчання.

Така проблема пов'язана з тим, що при малому кроці час навчання буде більшим і мережа може застрягати в локальних мінімумах, а при великих кроках можлива розбіжність процесу навчання або параліч мережі. Проблема ефективно вирішується адаптивним кроком, який для кожної ітерації дозволяє зробити крок, який мінімізує помилку мережі на даній ітерації.

Існують методи, які на кожному тренувальному циклі (званому епохою) аналізують всю тренувальну вибірку і вибирають оптимальне значення та напрямок кроку.

2.2.2 Нейронні мережі високого порядку та моментні НР

Нейронні мережі високого порядку (НМВП, High Order Neural Network) відрізняються від БНМ тим, що вони мають лише один шар, але на входи нейронів надходять також члени високого порядку, які є твором двох або більше компонентів вхідного вектора, наприклад: для мереж другого порядку $S = \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=1}^n w_{ij} x_i x_j - T$.

Такі мережі можуть формувати складні розділяючі поверхні.

Розділяюча поверхня другого порядку $S = 0$ називається гіперквадрикою. Додаючи компоненти вхідного вектора до твір, отримаємо клас поліноміальних поверхонь, що розділяють. Такі мережі також можна навчати методом зворотного поширення. Багатошарові НМ в загальному випадку ефективніші, але існує ряд додатків, в яких мережі високого порядку краще, ніж БНМ.

Крім того, така нейронна мережа має високу швидкість навчання та обробки інформації. Експериментально помічено, що нейронні мережі високого порядку мають підвищену точність ідентифікації відвернутих та масштабованих зображень, на відміну від БНМ.

2.2.3 Радіально-базисні нейронні мережі

Радіально-базисні нейронні мережі (РБНМ, Radial Basis Function Network, RBF) продовжують ідею карт Кохонена і складаються з двох шарів. Перший шар має радіально-базисну активаційну функцію:

$$y = \exp \exp \left(\frac{-S^2}{2\sigma^2} \right)$$

де σ – середньоквадратичне відхилення, характеризує ширину функції (розмір кластера).

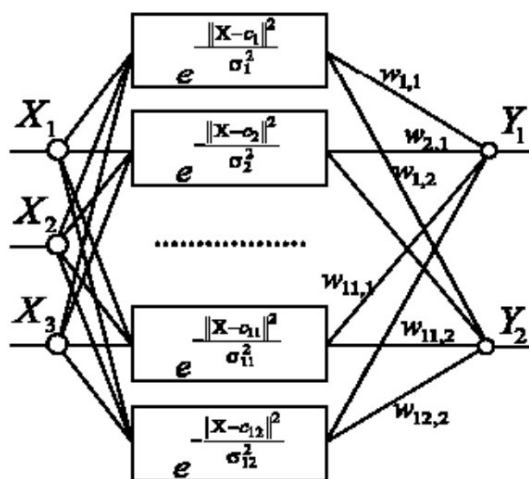


Рисунок 2.3 – Радіально-базисна нейронна мережа. X_i – елементи вхідного вектора, Y_i – елементи вихідного вектора, квадратами позначені радіально-базисні нейрони.

Таким чином, прихований шар є набором кластерів у просторі образів і виконує перший етап кластеризації вхідного образу – значення активаційної функції кожного нейрона швидко зменшується з видаленням від центру кластера. Другий шар нейронів має лінійну активаційну функцію та виконує другий етап кластеризації – розподіляє кластери за класами. На відміну від карт

Кохонена тут обнулення нейронів, що не володіють максимальним вихідним значенням, не потрібно, всі вони роблять внесок у класифікацію, і це перевага РБНМ.

РБНМ також здатна будувати складні розділяючі області та апроксимувати багатовимірні функції. У порівнянні з багатошаровою нейронною мережею, радіально-базова мережа навчається на порядок швидше, проте має набагато гірше екстра полірувальну здатність. Вона не здатна працювати на образах, що лежать далеко від образів прикладів. Розміри РБНМ більше, ніж МНС для аналогічних завдань, і РБНМ стають малоефективними із зростанням розмірності вхідних даних.

2.3 Теорія способів машинного навчання

2.3.1 Глибинне структуроване навчання

Глибинне навчання або глибинне структуроване навчання (англ. deep learning) є частиною ширшого сімейства методів машинного навчання, заснованих на штучних нейронних мережах із навчанням представленням. Навчання може бути контрольованим, напів контрольованим або неконтрольованим.

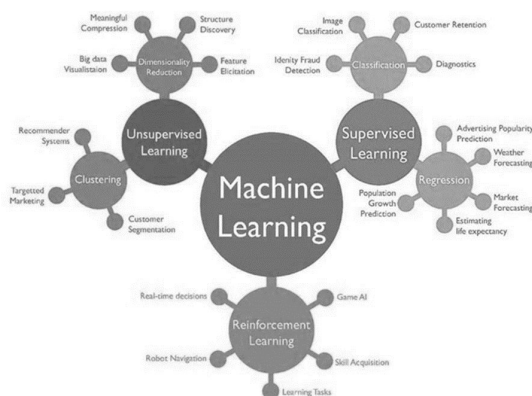


Рисунок 2.4 – Види способів машинного навчання

Архітектури глибокого навчання, такі як глибокі нейронні мережі, глибинні мережі переконань, глибинне навчання з підкріпленням, рекурентні нейронні мережі та згорткові нейронні мережі застосовувалися в таких областях, як комп'ютерне бачення, розпізнавання мови, обробка природної мови, машинний переклад, біоінформатика, розробка ліків, аналіз медичних зображень, кліматологія, інспекція матеріалів і програми настільних ігор, де вони дали результати, які можна порівняти з результатами експертів, а в деяких випадках навіть перевершити їх.

Прикметник «глибинний» у глибинному навчанні відноситься до використання кількох рівнів у мережі. Рання робота показала, що лінійний перцептрон не може бути універсальним класифікатором, але мережа з неполіноміальною функцією активації з одним прихованим шаром необмеженої ширини може.

Глибоке навчання — це сучасний варіант, який стосується необмеженої кількості шарів обмеженого розміру, що дозволяє практичне застосувати та оптимізувати впровадження, зберігаючи теоретичну універсальність у нормальних умовах. У глибокому навчанні шарами також дозволяється бути неоднорідними та значно відхилятися від біологічно інформованих конекціоністських моделей заради підвищення загальної ефективності, ефективності навчання та зрозумілості.

2.3.2 Кероване навчання

Кероване або контрольоване навчання (іноді «навчання з учителем», англ. Supervised learning) – це парадигма машинного навчання для задач, у яких доступні дані складаються з позначених прикладів, тобто кожна точка даних містить ознаки і пов'язану мітку. Метою алгоритмів контрольованого навчання є навчання функції, яка поєднувала б вектори ознак (вхідні дані) з мітками (вихідні) на основі прикладів пар введення-виведення. Кероване навчання виводить функцію з позначених навчальних даних, що складаються з набору навчальних прикладів. У керованому навчанні кожен приклад є парою, що складається з вхідного об'єкта (зазвичай вектора) і бажаного вихідного значення (також називається контрольним сигналом). Алгоритм навчання під наглядом аналізує навчальні дані та створює передбачувану функцію, яку можна використовувати для відображення нових прикладів. В ідеальному випадку алгоритм має правильно визначити мітки класу для невідомих екземплярів. Для цього потрібно, щоб алгоритм навчання узагальнив навчальні дані для невідомих прикладів «розумним» способом. Ця статистична якість алгоритму вимірюється через так звану помилку узагальнення.

2.3.3 Напівкероване навчання

Напівкероване навчання – це підхід до машинного навчання, який поєднує невелику кількість позначених даних із великою кількістю не позначених даних під час навчання. Напів контрольоване навчання знаходиться між неконтрольованим навчанням (без позначених навчальних даних) і контрольованим (тільки з позначеними навчальними даними). Це особливий випадок навчання зі слабким наглядом.

Дані без міток у поєднанні з невеликою кількістю мічених даних можуть значно підвищити точність навчання. Отримання мічених даних для навчання часто потребує зусиль кваліфікованого спеціаліста (наприклад, для транскрипції аудіо сегменту) або проведення фізичного експерименту (наприклад, визначення тривимірної структури білка або визначення наявності нафти в певному місці). В таких ситуаціях збір великої кількості фактичних даних є дуже дорогим або неможливим, в той час як отримання немаркованих даних є відносно дешевим. У таких ситуаціях напів контрольоване навчання може мати велике практичне значення через скорочення кінцевої вартості продукту.

Обробляються набори незалежно ідентично розподілених прикладів з відповідними мітками та прикладів без міток. Напів контрольоване навчання поєднує цю інформацію, щоб перевершити ефективність класифікації, яку можна отримати, відкинувши немічені дані та виконавши контрольоване навчання, або відкинувши мітки та виконавши неконтрольоване навчання.

Напівкероване навчання може стосуватися або трансдуктивного навчання, або індуктивного навчання. Мета трансдуктивного навчання полягає в тому, щоб вивести правильні мітки тільки для заданих немаркованих даних. Мета індуктивного навчання полягає в правильному відображенні між позначеними прикладами та мітками.

Інтуїтивно зрозуміло, що навчальну задачу можна розглядати як іспит, а дані позначати як приклади завдань, які вчитель розв'язує для класу як допоміжний засіб у розв'язанні іншого набору завдань. У трансдуктивній обстановці ці невирішені задачі грають роль екзаменаційних запитань. В індуктивній обстановці вони стають практичними завданнями, які складатимуть іспит.

Немає потреби (і, згідно з принципом Вапника, необачно) виконувати трансдуктивне навчання шляхом виведення правила класифікації по всьому простору введення; однак на практиці алгоритми, формально розроблені для трансдукції або індукції, часто використовуються як взаємозамінні.

2.3.4 Неконтрольоване навчання

Неконтрольоване навчання або навчання без вчителя (англ. Unsupervised learning) – це парадигма машинного навчання для задач, у яких доступні дані складаються з непозначених прикладів, тобто кожна точка даних містить лише ознаки без пов'язаної мітки. Метою алгоритмів неконтрольованого навчання є вивчення корисних шаблонів або структурних властивостей даних. Прикладами завдань навчання без контролю є кластеризація, зменшення розмірності та оцінка щільності.

Надія застосування неконтрольованого навчання полягає в тому, що за допомогою мімікрії, яка є важливим способом навчання людей, алгоритм машинного навчання навчиться створювати компактне внутрішнє представлення даних. У випадку генеративного завдання таке представлення може бути корисним і навіть необхідним для того, щоб алгоритм генерував з нього образний контент.

На відміну від навчання під наглядом, де дані маркуються експертом, неконтрольовані методи демонструють самоорганізацію, яка фіксує шаблони як щільність ймовірності або комбінацію уподобань нейронних особливостей. Іншими парадигмами навчання в спектрі нагляду є навчання з підкріпленням, коли машині дається лише числова оцінка продуктивності як орієнтир, і напівконтрольоване навчання, де позначається тільки менша частина даних.

2.3.5 Навчання з підкріпленням

Навчання з підкріпленням (англ. reinforcement learning) — це галузь машинного навчання, яка вивчає, як інтелектуальні агенти повинні виконувати дії в середовищі, щоб максимізувати поняття сукупної винагороди. Навчання з підкріпленням є однією з трьох основних парадигм машинного навчання, поряд із навчанням під наглядом і навчанням без нагляду.

Навчання з підкріпленням відрізняється від навчання під наглядом тим, що не потрібно подавати позначені пари введення/виходу, а також не потрібно явно виправляти субоптимальні дії. Натомість увага зосереджена на пошуку балансу між дослідженням (ще невідомих даних) і використанням (уже набутих знань).

Середовище зазвичай описується у формі марковського процесу прийняття рішень (МПВ – марковського процесу вирішення), тому що багато алгоритмів навчання з підкріпленням для цього контексту використовують методи динамічного програмування. Основна відмінність між класичними методами динамічного програмування та алгоритмами навчання з підкріпленням полягає в тому, що останні не передбачають знання точної математичної моделі МПВ, і вони націлені на великі МПВ, де точні методи стають неможливими.

2.3.6 Регресійний аналіз

Регресійний аналіз – це набір статистичних процесів для оцінки зв'язків між залежною змінною (яку часто називають змінною «результату» або «відповіді» або «міткою» на мові машинного навчання) та однією або кількома незалежними змінними (часто називають «предикторами», «коваріатами», «пояснювальними змінними» або «особливими ознаками»). Найпоширенішою формою регресійного аналізу є лінійна регресія, у якій знаходять лінію (або більш складну лінійну комбінацію), яка найбільше відповідає даним згідно з певним математичним критерієм.

Наприклад, метод найменших квадратів обчислює унікальну лінію (або площину), яка мінімізує суму квадратів різниць між справжніми даними та цією лінією (або площиною). З конкретних математичних міркувань це дозволяє досліднику оцінити умовне очікування (або середнє значення сукупності) залежної змінної, коли незалежні змінні приймають заданий набір значень. Менш поширені форми регресії використовують дещо інші процедури для оцінки альтернативних параметрів розташування (наприклад, квантильна регресія або аналіз необхідних умов) або оцінки умовного очікування для ширшої колекції нелінійних моделей (наприклад, непараметрична регресія).

Регресійний аналіз в основному використовується для двох концептуально різних цілей.

По-перше, регресійний аналіз широко використовується для передбачення та прогнозування, де його використання суттєво збігається з машинним навчанням.

По-друге, у деяких ситуаціях регресійний аналіз можна використовувати для висновку про причинно-наслідкові зв'язки між незалежними та залежними змінними. Важливо, що регресії самі по собі виявляють лише зв'язки між залежною змінною та набором незалежних змінних у фіксованому наборі даних.

Щоб використовувати регресії для прогнозування або для висновку про причинно-наслідкові зв'язки, відповідно, дослідник повинен ретельно обґрунтувати, чому існуючі зв'язки мають прогностичну силу для нового контексту або чому зв'язок між двома змінними має причинно-наслідкову інтерпретацію. Останнє особливо важливо, коли дослідники сподіваються оцінити причинно-наслідкові зв'язки, використовуючи дані спостережень.

2.3.7 Навчання на прикладах

Навчання на прикладах (іноді «навчання на основі пам'яті») – це група алгоритмів навчання, які замість виконання явного узагальнення порівнюють нові приклади проблеми з тими прикладами, що були помічені під час навчання, які зберігаються в пам'яті. Оскільки обчислення відкладається до появи нового екземпляра, ці алгоритми іноді називають «ледачими».

Метод навчання будує гіпотези безпосередньо з самих прикладів навчання. Це означає, що складність гіпотези може зростати разом із даними: у гіршому випадку гіпотеза — це список із n навчальних елементів, а обчислювальна складність класифікації одного нового екземпляра дорівнює $O(n)$. Однією з переваг методу навчання на основі прикладів перед іншими методами машинного навчання є його здатність адаптувати свою модель до раніше невідомих даних. Моделі, які вивчають приклади, можуть просто зберегти новий приклад або викинути старий.

Прикладами алгоритмів навчання на основі цього методу є алгоритм k -найближчих сусідів, ядрові методи та мережі радіально базисних функцій. Вони зберігають свій навчальний набір; коли прогнозують значення/клас для нового прикладу, вони обчислюють відстані або подібності між цим прикладом і навчальними прикладами, щоб прийняти рішення.

Для боротьби з підвищенням складності пам'яті, що пов'язана зі збереженням усіх прикладів навчання, а також із ризиком перенавчання, були запропоновані алгоритми зменшення кількості прикладів.

ВИСНОВКИ ДО РОЗДІЛУ 2

У ході роботи над цим розділом були детально проаналізовані необхідні підходи до алгоритмів ідентифікації об'єктів задля подальшого їх використання у роботі з пошуком воєнної техніки на знімках супутників. З аналізу обраної літератури та власних знань було зроблено висновки про доцільність використання різноманітних способів написання та навчання нейронної мережі і обрані найбільш актуальні для заданої теми роботи.

Вдалою та зручною для обробки зображень і пошуку відповідних об'єктів було визначено архітектуру загорткової НМ, яка дозволяє алгоритмам максимально ефективно розпізнавати образи. Сама ідея ґрунтується на чергуванні загорткових та субдискретизуючих шарів, а структура є односпрямованою, що ідеально підходить під задачі роботи.

Робота над розділом допомогла визначитися з послідовністю дій та їх направленістю для досягнення поставленої мети у написанні найбільш оптимізованої та раціональної програми, що ідентифікувати воєнну техніку і аналізувати зображення місцевості. Крім цього, було придумано декілька теоретичних розрахункових матеріалів, які теж знадобляться у розробці самої програми.

Крім цього необхідно було знайти відповідь на питання, яким саме чином навчати нейромережу буде найбільш оптимально і раціонально. Перш за все, було розглянуто деякі з можливих сучасних найбільш розповсюджених варіантів навчання – глибинне структуроване навчання, кероване навчання, напів кероване навчання, неконтрольоване навчання, регресійний аналіз, навчання на прикладах. Кожен з них має свої переваги та недолік у різних задачах, для використання був обраний вправний синтез для найбільшої оптимізації зусиль.

РОЗДІЛ 3. ОПИС РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ВИКОРИСТАНИХ ТЕХНОЛОГІЙ

3.1 Спосіб ідентифікації техніки та його застосування

Метою цієї роботи було створення алгоритму, що дозволить нам за допомогою супутникових зображень високої роздільної здатності знаходити та виділяти техніку, щоб у майбутньому використовувати для відстеження переміщення військ, перевірки атаки у недоступних районах та інше.

У якості застосування даного способу було реалізовано систему машинного навчання на основі датасетів військової техніки. Було використано алгоритм машинного навчання з учителем.

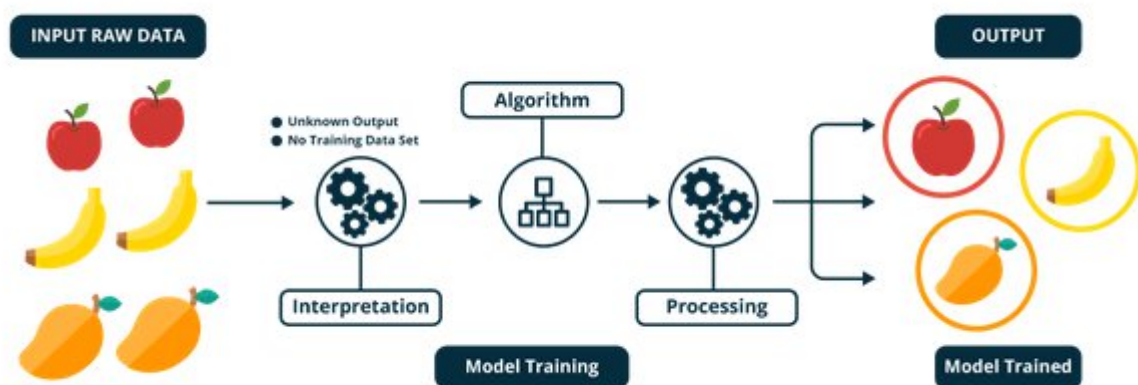


Рисунок 3.1 - Схема машинного навчання з учителем

Система працює наступним чином:

- береться датасет, який вже розділено на типи, та ділиться на тестові та перевірочні данні для моделі;
- датасети загружаються у модель;
- після завершення процесу навчання модель перевіряється на основі тестових даних (підмножина навчального набору), а потім прогнозується результат.

Для визначення найбільш оптимального методу ідентифікації техніки було обрано кілька алгоритмів для тестування, а саме:

- multilayer perceptron;
- згорткова нейронна мережа;
- оптимізована згорткова мережа.

3.1.2 Multilayer perceptron

Багатошаровий персептрон (MLP) — це повністю зв'язаний клас штучної нейронної мережі (ANN) із прямою передачею даних. Термін MLP використовується неоднозначно, іноді вільно для позначення будь-якої прямої ШНМ, іноді суворо для позначення мереж, що складаються з кількох рівнів персептронів (з пороговою активацією); див. § Термінологія. Багатошарові персептрони іноді в просторіччі називають «вільними» нейронними мережами, особливо коли вони мають один прихований шар.

MLP складається щонайменше з трьох рівнів шарів: вхідного, прихованого та вихідного. За винятком вхідних вузлів, кожен вузол є нейроном, який використовує нелінійну функцію активації. MLP використовує методику навчання під наглядом, яка називається зворотним поширенням для навчання. Його кілька шарів і нелінійна активація відрізняють MLP від лінійного персептрона. Він може розрізняти дані, які не є лінійно роздільними.

Функція активації:

Якщо багатошаровий персептрон має лінійну функцію активації в усіх нейронах, тобто лінійну функцію, яка відображає зважені входи на вихід кожного нейрона, тоді лінійна алгебра показує, що будь-яку кількість шарів можна звести до дворівневого входу. вихідна модель. У MLP деякі нейрони використовують нелінійну функцію активації, яка була розроблена для моделювання частоти потенціалів дії або активації біологічних нейронів.

Дві історично поширені функції активації є сигмовидними і описуються:

$$y(v_i) = \tanh(v_i) \text{ and } y(v_i) = (1 + e^{-v_i})^{-1}.$$

В останніх розробках глибинного навчання лінійний випрямляючий блок (ReLU) частіше використовується як один із можливих способів подолання численних проблем, пов'язаних із сигмоїда.

Перший є гіперболічним тангенсом, який змінюється від -1 до 1, тоді як інший є логістичною функцією, яка подібна за формою, але змінюється від 0 до 1. Тут

u_i є результатом i -го вузла (нейрон) і v_i — зважена сума вхідних з'єднань. Були запропоновані альтернативні функції активації, включаючи функції випрямляча та softplus. Більш спеціалізовані функції активації включають радіальні базисні функції (використовуються в радіальних базисних мережах, іншому класі контрольованих нейронних мережевих моделей).

Навчання:

Навчання відбувається в персептроні шляхом зміни вагових коефіцієнтів з'єднання після обробки кожного фрагмента даних на основі кількості помилок у виводі порівняно з очікуваним результатом. Це приклад навчання під наглядом, який здійснюється через зворотне поширення, узагальнення алгоритму найменших середніх квадратів у лінійному персептроні.

Ми можемо представити ступінь помилки у вихідному вузлі j у n -й точці даних (навчальний приклад) за допомогою $e_j(n) = d_j(n) - y_j(n)$, де d_j — цільове значення, а y_j — значення, створене персептроном. Потім ваги вузлів можна скоригувати на основі виправлень, які мінімізують помилку в усьому виведенні, заданому

$$\mathcal{E}(n) = \frac{1}{2} \sum_j e_j^2(n).$$

Використовуючи градієнтний спуск, зміна кожної ваги є

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathcal{E}(n)}{\partial v_j(n)} y_i(n)$$

де y_i — вихід попереднього нейрона, а η — швидкість навчання, яка вибирається для того, щоб вагові коефіцієнти швидко збігалися з відповіддю без коливань.

Похідна, яку потрібно обчислити, залежить від індукованого локального поля v_j , яке саме по собі змінюється. Легко довести, що для вихідного вузла цю похідну можна спростити до

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = e_j(n) \phi'(v_j(n))$$

де ϕ є похідною функції активації, описаної вище, яка сама по собі не змінюється. Аналіз є складнішим для зміни ваг до прихованого вузла, але можна показати, що відповідна похідна

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = \phi'(v_j(n)) \sum_k -\frac{\partial \mathcal{E}(n)}{\partial v_k(n)} w_{kj}(n).$$

Це залежить від зміни ваг k-х вузлів, які представляють вихідний рівень. Таким чином, щоб змінити ваги прихованого шару, ваги вихідного шару змінюються відповідно до похідної функції активації, і тому цей алгоритм представляє зворотне поширення функції активації.

Нижче, на рисунку 3.2 продемонстровано створення моделі на основі даної технології.

```
model_MLP = keras.Sequential()
model_MLP.add(InputLayer(input_shape=(image_size + (1,))))

model_MLP.add(Flatten())

model_MLP.add(Dense(100, activation='relu'))
model_MLP.add(Dense(100, activation='relu'))
model_MLP.add(Dense(100, activation='relu'))

model_MLP.add(Dense(num_classes, activation='softmax'))

model_MLP.summary()
```

Model: "sequential_10"

Layer (type)	Output Shape	Param #
flatten_7 (Flatten)	(None, 16384)	0
dense_26 (Dense)	(None, 100)	1638500
dense_27 (Dense)	(None, 100)	10100
dense_28 (Dense)	(None, 100)	10100
dense_29 (Dense)	(None, 9)	909

=====
Total params: 1,659,609
Trainable params: 1,659,609
Non-trainable params: 0

Рисунок 3.2 – Код моделі MLP

3.1.3 Згорткова нейронна мережа

Згорткова нейронна мережа (CNN або convnet) є підмножиною машинного навчання. Це один із різних типів штучних нейронних мереж, які використовуються для різних програм і типів даних. CNN — це свого роду мережева архітектура для алгоритмів глибокого навчання, яка спеціально використовується для розпізнавання зображень і завдань, які включають обробку піксельних даних.

Існують інші типи нейронних мереж у глибокому навчанні, але для ідентифікації та розпізнавання об'єктів CNN є мережевою архітектурою вибору. Це робить їх дуже придатними для завдань комп'ютерного зору (CV) і для програм, де розпізнавання об'єктів є життєво важливим, наприклад, для самокерованих автомобілів і розпізнавання облич.

Нижче продемонстровано створення моделі на основі згорткової нейронної мережі.

```
model_CNN = tf.keras.Sequential([
    tf.keras.layers.InputLayer(input_shape=(image_size + (1,))),
    tf.keras.layers.Rescaling(1./255),
    tf.keras.layers.Conv2D(32, 3, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(32, 3, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(32, 3, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dense(num_classes, activation='softmax')
])
model2.summary()
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 128, 128, 1)	0
conv2d (Conv2D)	(None, 126, 126, 32)	320
max_pooling2d (MaxPooling2D)	(None, 63, 63, 32)	0
conv2d_1 (Conv2D)	(None, 61, 61, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 30, 30, 32)	0
conv2d_2 (Conv2D)	(None, 28, 28, 32)	9248
max_pooling2d_2 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten_1 (Flatten)	(None, 6272)	0
dense_4 (Dense)	(None, 128)	802944
dense_5 (Dense)	(None, 9)	1161

=====
 Total params: 822,921
 Trainable params: 822,921
 Non-trainable params: 0

Рисунок 3.3 – Створення моделі згорткової нейронної мережі

3.2 Засоби розробки

Будь яка робота, у тому числі й розробка ПЗ вимагає певного набору інструментів. Під цим ми розуміємо набір програмного забезпечення, який допоможе нам в розробці нашого проекту.

Під час роботи над дипломною роботою було обрано наступні засоби розробки:

- jupyter notebook;
- git;
- tensorflow;
- keras;
- matplotlib;
- numpy;
- sklearn.

Особливо треба виділити tensorflow та keras.

TensorFlow — це open-source платформа з відкритим вихідним кодом для створення програм машинного навчання. Це символічна математична бібліотека, яка використовує потік даних і диференційоване програмування для виконання різноманітних завдань, зосереджених на навчанні та висновках глибоких нейронних мереж. Це дозволяє розробникам створювати програми машинного навчання за допомогою різних інструментів, бібліотек і ресурсів спільноти. Зараз найвідомішою бібліотекою глибокого навчання у світі є TensorFlow від Google. Google використовує машинне навчання в усіх своїх продуктах, щоб покращити пошукову систему, переклад, підписи до зображень або рекомендації.

Keras — це API глибокого навчання, написаний на Python, який працює на платформі машинного навчання TensorFlow. Він був розроблений з упором на можливість швидкого експериментування.

3.3 ДАТАСЕТ

Для цього проекту я використовував набір даних для виявлення та розпізнавання рухомих і стаціонарних цілей (MSTAR), створений Агентством передових оборонних дослідницьких проектів США (DARPA) і Дослідницькою лабораторією ВПС США з 1995 по 1997 рік. Набір даних доступний для загального використання на веб-сайті науково-дослідної лабораторії ВПС. Вам потрібно буде створити обліковий запис, щоб завантажити його. Дані містять такі 9 класів:

- 1) 2С1 Гвоздика — Самохідна артилерійська установка;
- 2) ЗСУ-23—4 Шилка — зенітна самохідна установка;
- 3) БРДМ-2 — броньована машина-розвідник-амфібія;
- 4) БТР-60 — бронетранспортер;
- 5) D7 — Бульдозер Caterpillar;
- 6) ЗІЛ-131 — Військово-вантажний автомобіль;
- 7) Т-62 — Основний бойовий танк;
- 8) Т-72 — основний бойовий танк 2-го покоління;
- 9) SLICY.

Дозвіл на використання даних було запрошено на сайті даного датасету.

Перед використанням, дані треба було обробити, тому було зроблено наступне:

```
train_dataset = image_dataset_from_directory(
    'mstar_images',
    subset='training',
    image_size=image_size,
    labels='inferred',
    validation_split=.2,
    seed=10,
    label_mode='categorical',
    color_mode='grayscale',
    batch_size=batch_size,
)

val_dataset = image_dataset_from_directory(
    'mstar_images',
    subset='validation',
    image_size=image_size,
    labels='inferred',
    validation_split=.2,
    seed=10,
    label_mode='categorical',
    color_mode='grayscale',
    batch_size=batch_size,
)

Found 4333 files belonging to 9 classes.
Using 3467 files for training.
Found 4333 files belonging to 9 classes.
Using 866 files for validation.
```

Рисунок 3.4 – Запит на використання даних

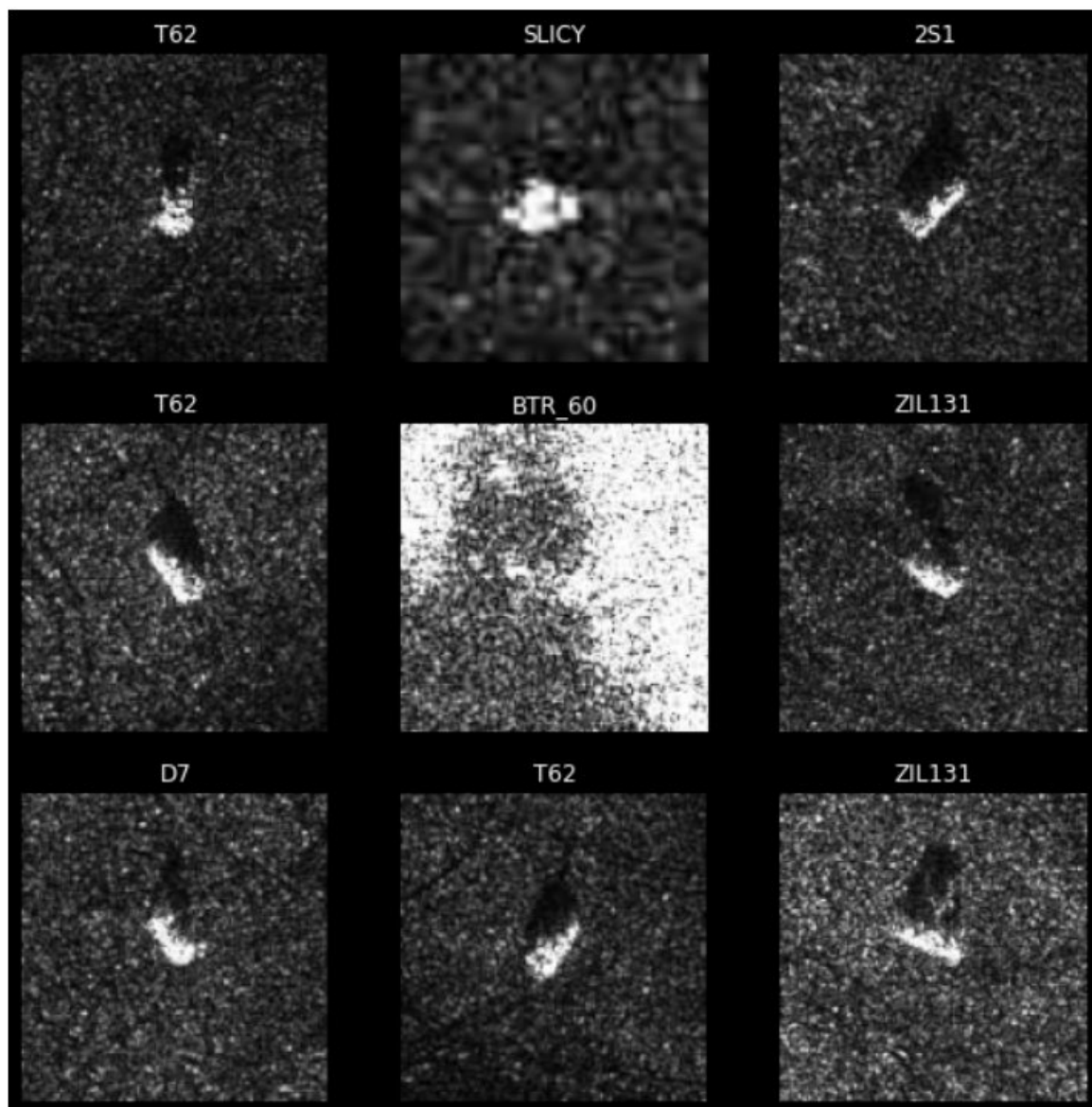


Рисунок 3.5 – Використання запитаних даних

3.3.1 Метрики оцінювання

Як я вже казав, метою цього проекту було створити модель, яка була б максимально наближена до ідеального класифікатора. Щоб сформулювати напрямок моєї роботи, я використовував пару показників, агрегованих у 9 класах, щоб керувати моїми ітераціями:

- категорична точність: середнє значення показників точності для 9 класів;
- середньозважена оцінка F2: для цього випадку використання я хотів надати більший пріоритет запам'ятовуванню над точністю, тому я використав середню оцінку F2 за категоріями;
- Коефіцієнт кореляції Метьюза: числове представлення ефективності класифікатора. 1 = Ідеально, 0 = Випадкове вгадування. Я використовував це як показник North Star, який інформував про зміни моделі.

Результати практичних іспитів

Нижче будуть перераховані результати та метрики тестових моделей, а також представлена остання модель. Її основним модифікуванням в порівнянні з типічною CNN моделлю буде розширення даних і шарів відсіву. Також було додано два параметри зворотного виклику: рання зупинка та зниження швидкості навчання на плато. Рання зупинка була зроблена для припинення навчання моделі, якщо відстежуванні показники не покращуються протягом певної кількості епох. Тим часом зниження швидкості навчання на плато призведе до зниження швидкості навчання моделі, якщо не буде покращення протягом певної кількості епох. Важливо встановити, щоб швидкість навчання була нижчою за швидкість ранньої зупинки, щоб модель мала можливість спробувати нову швидкість перед зупинкою.

```
my_model = tf.keras.Sequential([
    tf.keras.layers.InputLayer(input_shape=(image_size + (3,))),
    tf.keras.layers.Rescaling(1./255),
    tf.keras.layers.RandomFlip('horizontal_and_vertical'),
    tf.keras.layers.RandomRotation(0.2),
    tf.keras.layers.RandomZoom(0.1),
    tf.keras.layers.Conv2D(128, 5, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(64, 4, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(32, 3, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(16, 2, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(256, activation='relu'),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(num_classes, activation='softmax')
])
```

Рисунок 3.6 – Модель розширення даних і шарів відсіву

Нижче на рисунках 3.7, 3.8 показані результати для Multilayer perceptron, далі на 3.9, 3.10 для згорткової нейронної мережі, і 3.11 та 3.12 – модифікована згорткова нейронна мережа.

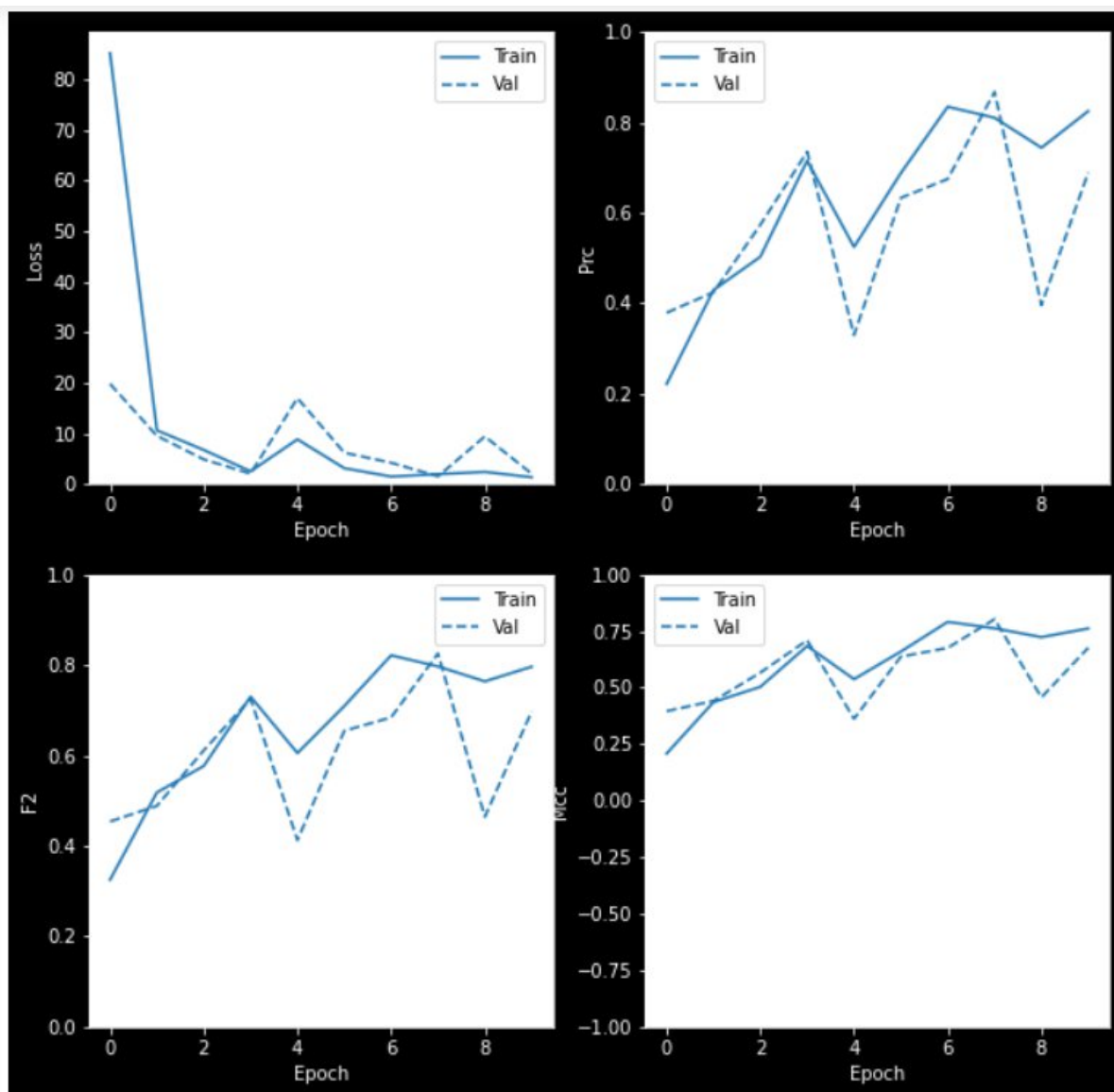


Рисунок 3.7 – Результати для Multilayer perceptron

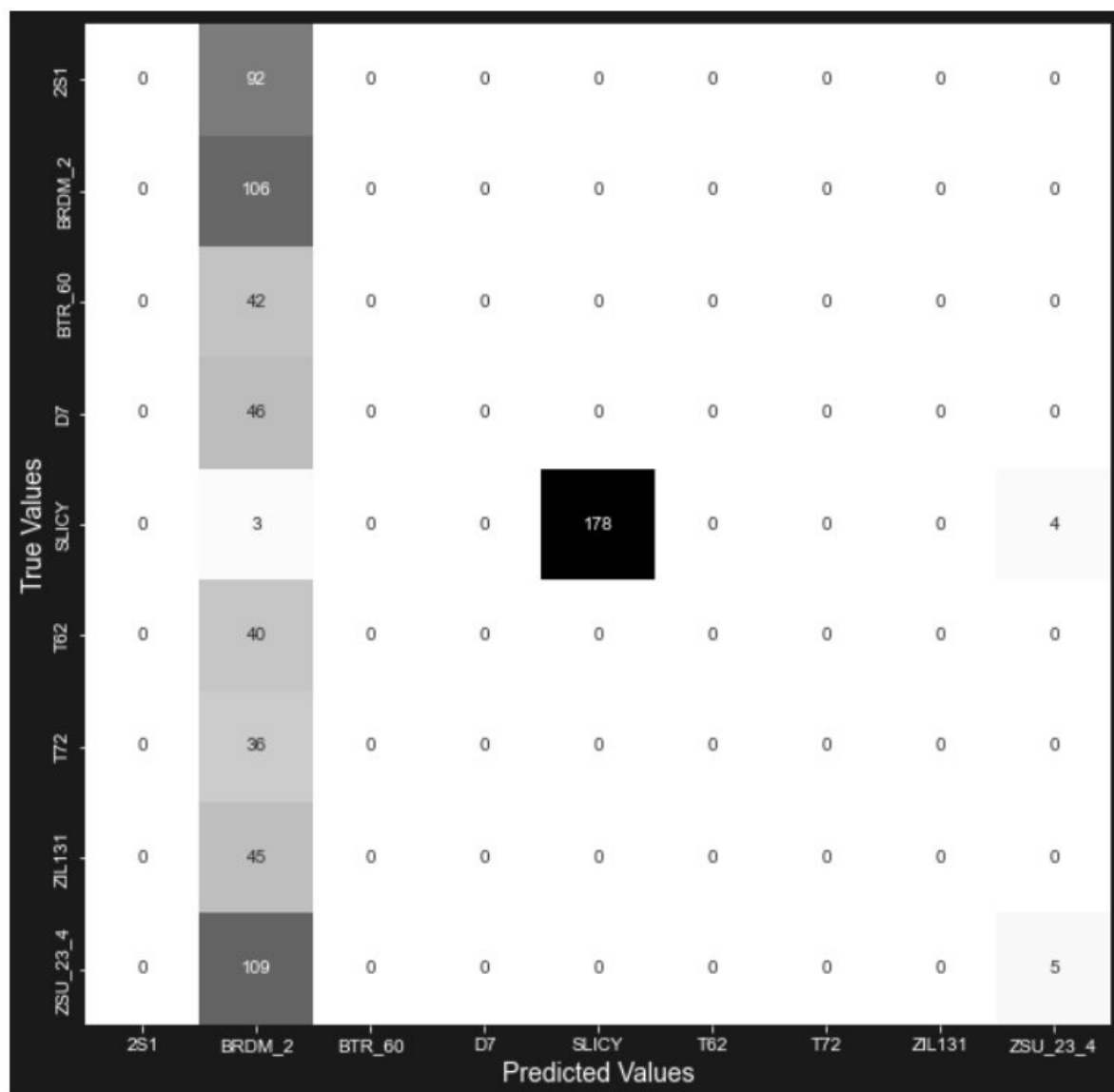


Рисунок 3.8 – Результаты для Multilayer perceptron

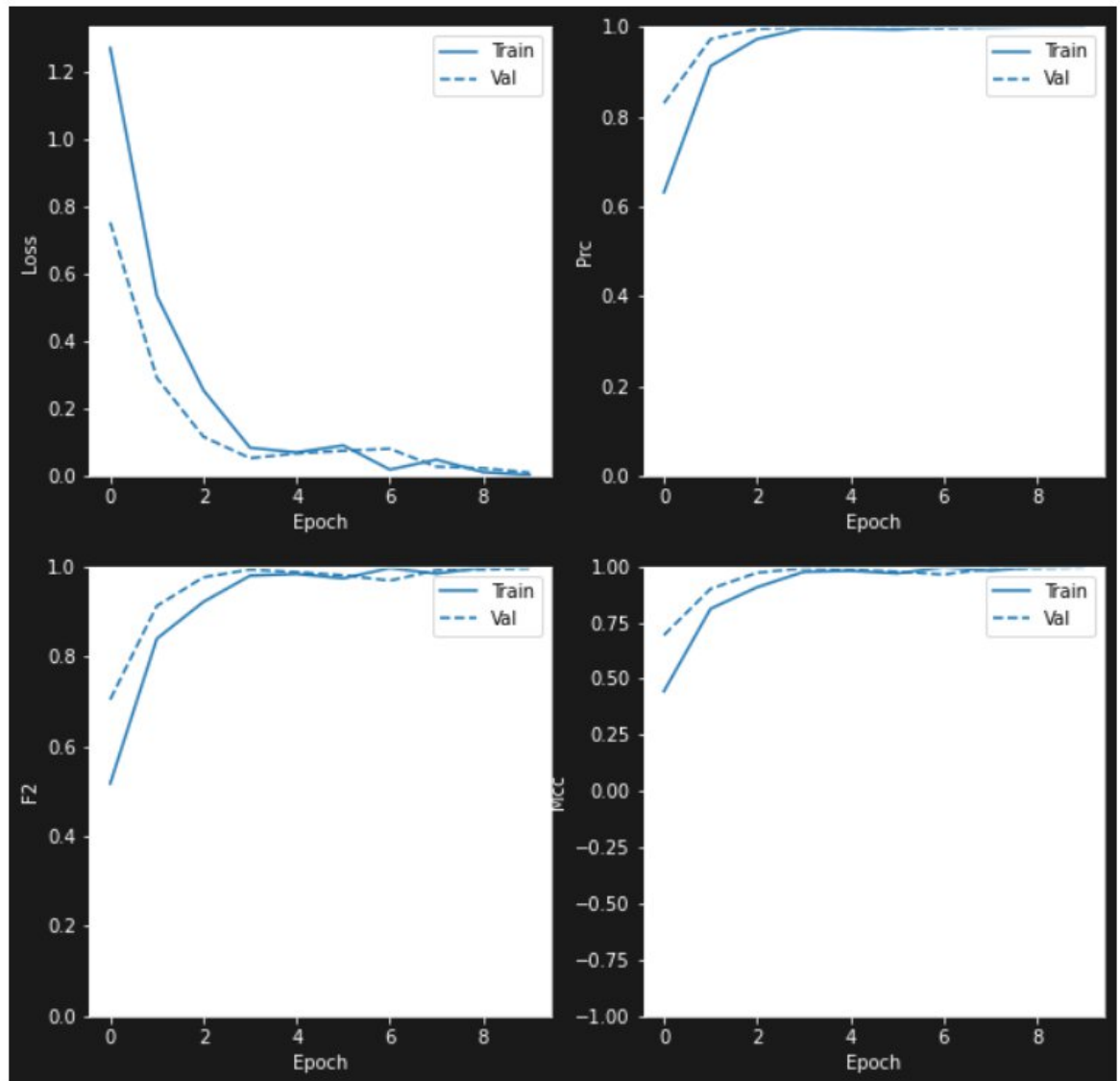


Рисунок 3.9 – Результати для згорткової нейронної мережі

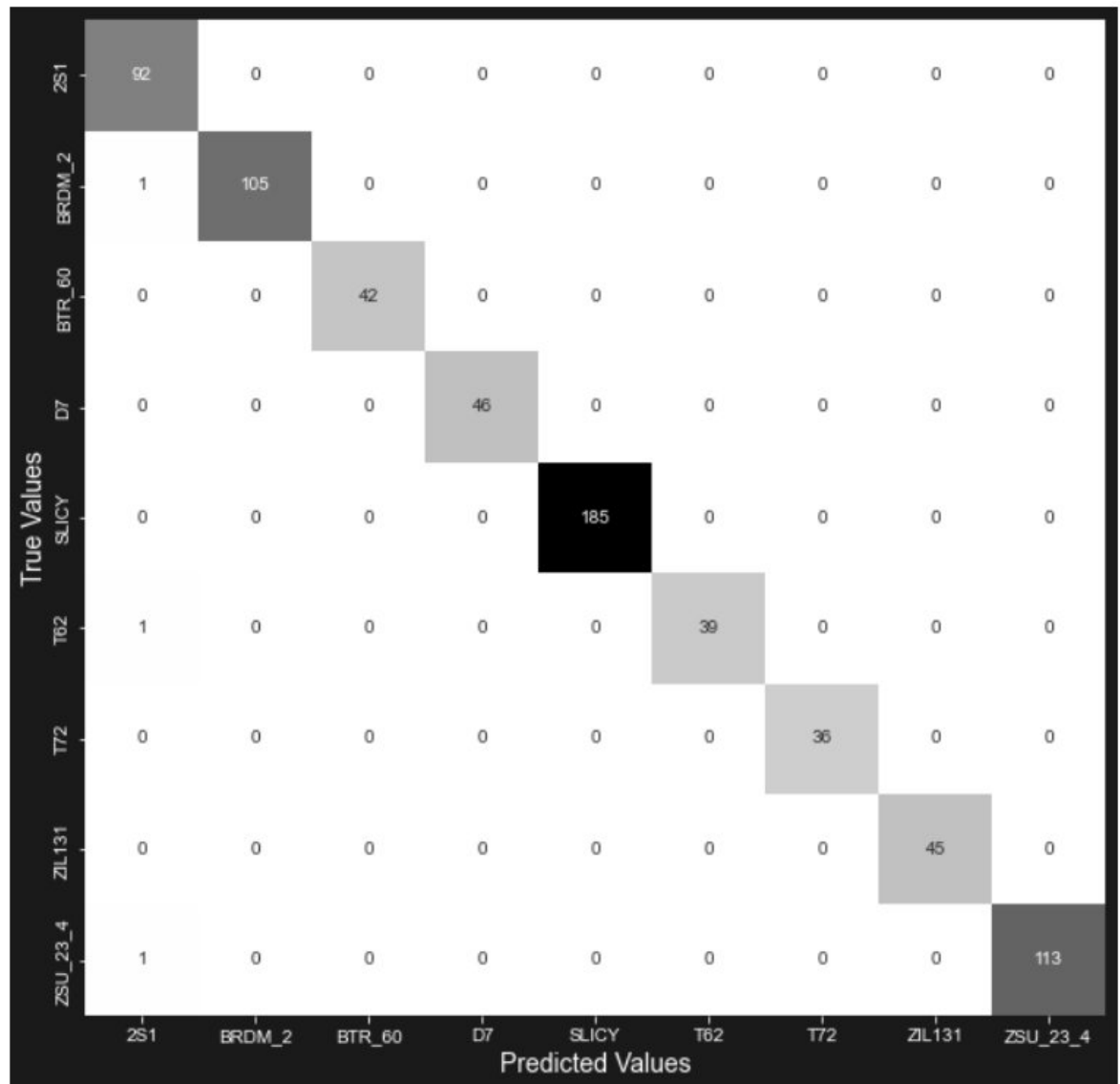


Рисунок 3.10 – Результати для згорткової нейронної мережі

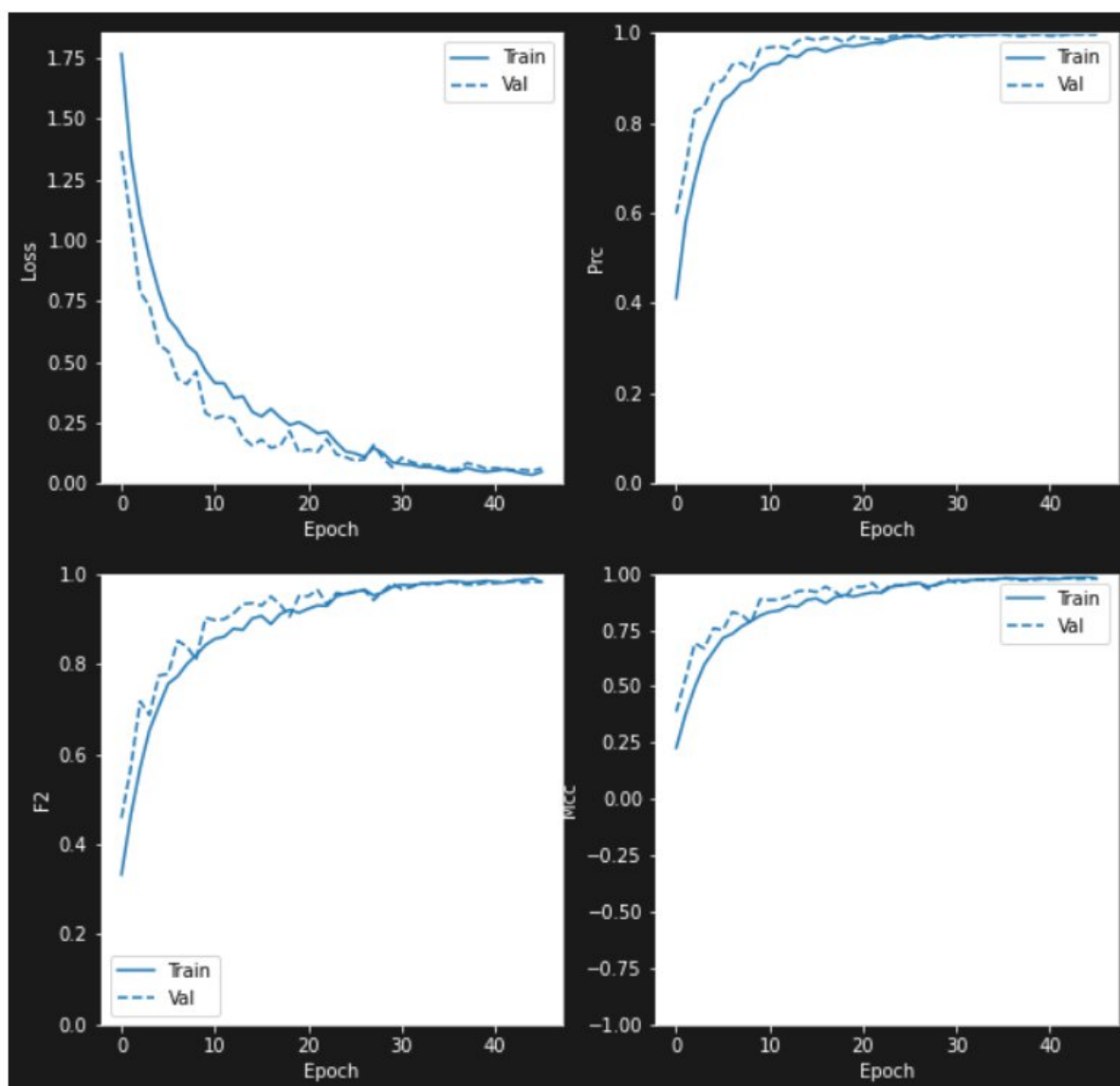


Рисунок 3.11 – Модифікована згорткова нейронна мережа

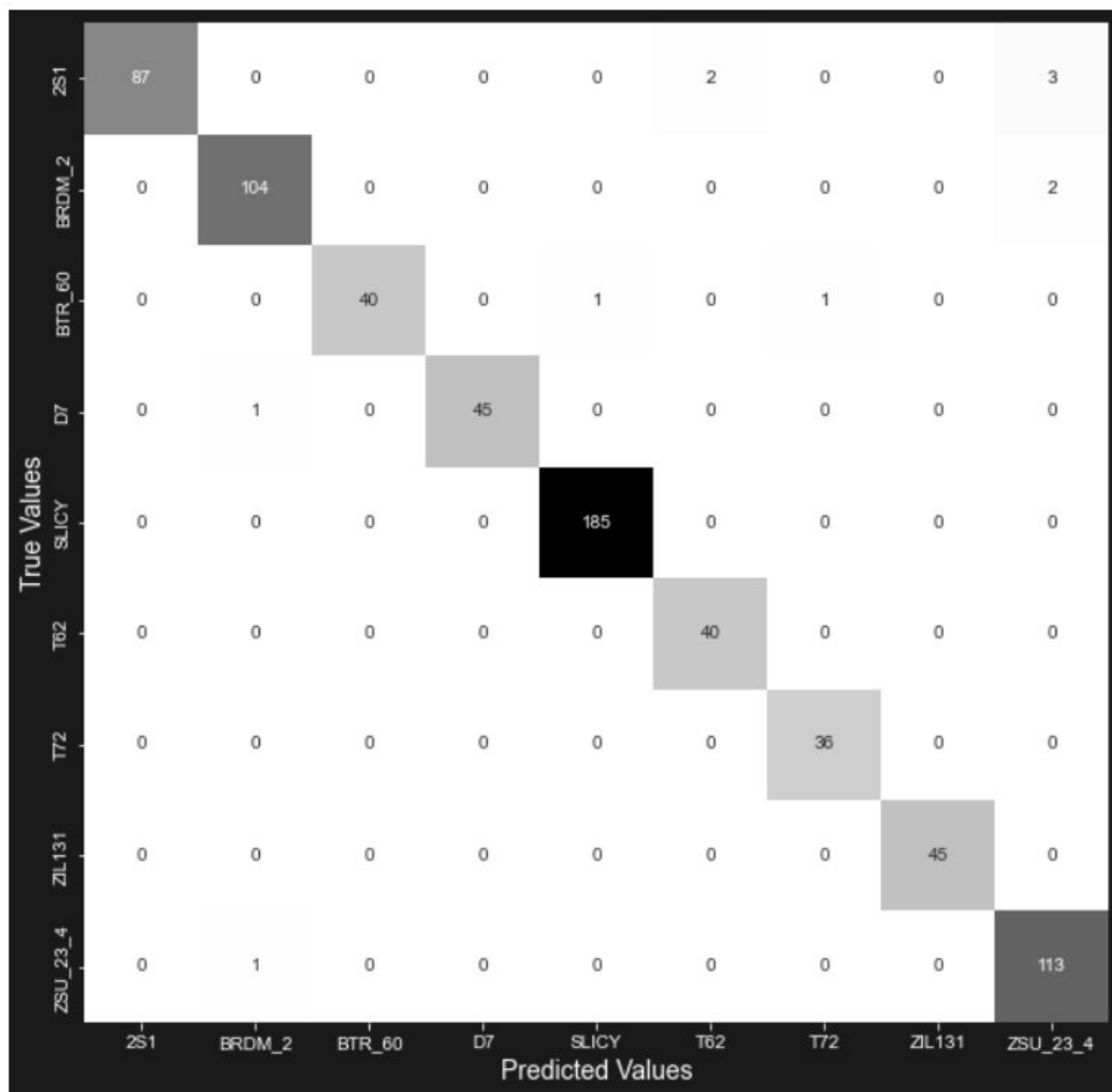


Рисунок 3.12 – Модифікована згорткова нейронна мережа

Як ми побачили вище, алгоритми мають різну точність виявлення типів техніки на основі датасету. Практика показує, що усе ж таки найбільш оптимальними є алгоритми CNN, та їм подібні.

На рисунку 3.13 показано логічна модель створеної згорткової нейронної мережі.

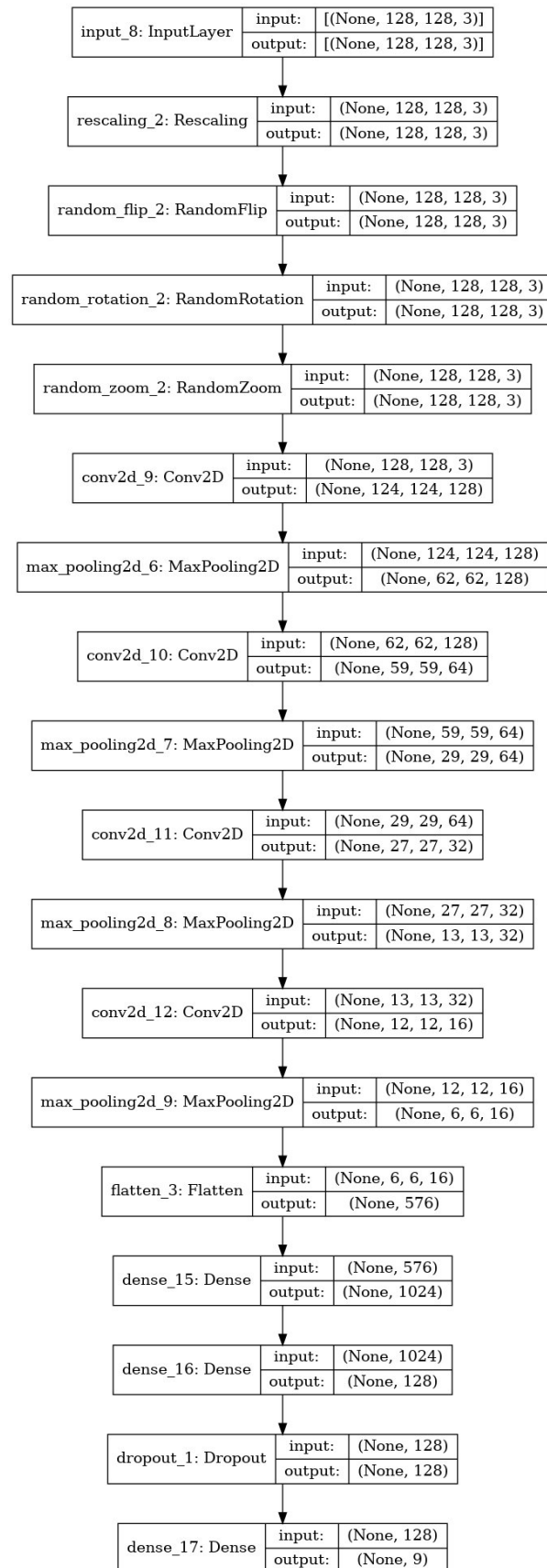


Рисунок 3.13 – Логічна модель створеної згорткової нейронної мережі.

ВИСНОВКИ ДО РОЗДІЛУ 3

У третьому розділі було розкрито логіку деяких алгоритмів машинного навчання. Було описано які данні використовувались у розробці, та як валідувались. Також було описано програмне забезпечення моделей та показані метрики на основі даних з датасету. Метрики вказують, що алгоритми CNN подібні більш підходять для виявлення типів техніки на знімках із супутника. Також були проведені тести на даних не з датасету, і було виявлено, що за рамками даних MSTAR моделі не можуть точно визначати техніку за її типами.

РОЗДІЛ 4. РОЗРОБКА СТАРТАП-ПРОЕКТУ

4.1 Опис ідеї проекту

Опис ідеї стартап-проекту, напрямки застосування ідентифікації описані у таблиці 4.1.

Таблиця 4.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Розробка системи оперативної ідентифікації воєнної техніки на знімках із супутника	1. Ідентифікація воєнної техніки у мирний чи воєнний час	Розуміння наявності воєнної техніки на знімках із супутника
	2. Ідентифікація типів воєнної техніки під час бойових дій	Розуміння типів воєнної техніки на знімках із супутника
	3. Ідентифікація воєнних споруд	Розуміння наявності воєнних споруд які можуть використовуватися під час бойових дій

Система оперативної ідентифікації воєнної техніки на знімках із супутника дозволяє отримати інформацію о наявності техніки та воєнних споруд на знімках, та отримувати інформацію о типах техніки, та споруд.

Довгостроковими перспективами на розвиток є:

- додавання системи розпізнавання об'єктів;
- додавання системи підпису об'єктів на фото;
- збільшення даних для моделі;
- додавання розпізнавання техніки в реальному часі.

Потреби в стартовому фінансуванні:

Стартовий капітал = 30000 грн

У таблиці 4.2 наведено техніко-економічні характеристики даної ідеї.

Таблиця 4.2 – Визначення сильних, слабких та нейтральних характеристик ідеї розробки проекту ідентифікації воєнної техніки на знімках із супутника.

№ з / п	Техніко Економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Конкурент 1 Microsoft	Конкурент 2 Google	Конкурент 3 Tesla			
1.	Бюджетне фінансування	власні кошти	розробка комерційна	розробка комерційна	розробка комерційна	Відсутність будь-якого фінансування	часткова бюджетне фінансування	виділення бюджетних коштів на розробку
2.	Використання сучасної техніки	використання сучасного обладнання	використано найсучаснішу техніку для проектування	використано найсучаснішу техніку для проектування	використано найсучаснішу техніку для проектування	використано застарілу техніку	часткова комплектація технікою	використано найсучаснішу техніку та технічні засоби для проектування
4.	Налагоджена система реклами продукту	реклама відсутня	реклама присутня	реклама присутня	реклама присутня	немає реклами	часткова реклама	реклама присутня
5.	Високий рівень розробки	запропоновані методи та алгоритми є досконалими	запропоновані методи та алгоритми є досконалими	Запропоновані методи та алгоритми є досконалими	запропоновані методи та алгоритми є досконалими	на базі розробки є помилки вона потребує доробок	розробка є майже досконалою	запропоновані методи та алгоритми є досконалими
6.	Професіонали програмісти	Розробником є студент	Головний розробник - група професіоналів	Головний розробник - група професіоналів	Головний розробник - група професіоналів	Головний розробник - студент	Головний розробник – професіонал	Головний розробник - група професіоналів

Технологічний аудит ідеї данного проекту :

Визначення технологічної здійсненності ідеї проекту передбачає аналіз таких складових у таблиці 4.3:

- яка технологія буде використана для розробки продукту;
- чи існують схожі технології, або їх потрібно розробити/додати?
- чи є в вільному доступі дані технології.

Таблиця 4.3 – Технологічна здійсненність ідеї проекту розробки інформаційної системи планування ресурсів IT-проектів

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Особливість і ідентифікації техніки на знімках з супутника за допомогою алгоритмів машинного навчання	Технологія 1 (алгоритм машинного навчання)	реалізована	доступні
2		Технологія 2 (наявність бази для навчання)	наявні, треба більше	доступні
3		Технологія 3 (модель для навчання)	реалізована/ треба доробити	доступні
4		Технологія 4 (оформлення результатів дослідження)	потрібно розробити	доступні
Обрана технологія ідентифікації техніки за допомогою машинного навчання є реалізуємо але потребує більше даних для навчання.				

4.3 Аналіз ринкових можливостей запуску стартап проекту

Визначення можливостей, які можна використовувати під час ринкового впровадження проекту, та ринкових загроз, явних та неявних, що можуть завадити реалізації проекту, дає нам можливість планувати напрями розвитку проекту з урахуванням стану ринкового середовища на сьогодні, потреб потенційних клієнтів та технології проектів-конкурентів. Цю інформацію представлено у таблиці 4.4.

Таблиця 4.4 – Стартова характеристика потенційного ринку для даного стартап проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	4
2	Загальний обсяг продаж, грн/ум.од	100000
3	Динаміка ринку (якісна оцінка)	Зростає(зараз)
4	Наявність обмежень для входу (вказати характер обмежень)	Тематика проекту, та його можливе використання
5	Середня норма рентабельності в галузі (або по ринку), %	33

Проведене дослідження показує, що ми можемо стверджувати про рентабельність проекту, та привабливість входження на ринок.

У таблиці нижче, 4.5 наведено перелік характеристик для потенційних клієнтів даного стартап-проекту, далі у таблиці 4.6 буде наведено фактори загроз й можливостей відповідно. У таблиці 4.7 буде наведено ступеневий аналіз конкуренції на ринку зараз.

Таблиця 4.5 – Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреби ринку	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних груп клієнтів	Вимоги споживачів до товару
1	Інформація про наявність та пересування воєнної техніки в мирний час та під час бойових дій, ділення техніки за типами	Державна оборона України, ЧВК	Державна оборона зможе оперативна ідентифікувати ворожі обсяги, їх типи та знаходження, коли ЧВК можуть використовувати у своїх комерційних цілях	Обов'язкове тестування на реальних даних

Таблиця 4.6 – Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Відсутність реальних даних для тренування	Більша погрішність для виявлення типів техніки на реальних даних	Більша погрішність для виявлення типів техніки на реальних даних
2	Висока вартість створення датасетів	Збільшення ціни на підтримку систему	Пошук більш дешевих аналогів
3	Економічні складності при розробці	відсутність фінансування повна чи часткова, або мінімальне фінансування	Пошук більш дешевих аналогів
4	Нестабільність політичної ситуації як у світі так і на території України	Неможливість у нормальних умовах розробляти продукт	Пошук способів для більш продуктивної та реальної розробки

Таблиця 4.7 – Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Тривале існування системи відстеження техніки	Тривале існування та забезпечення розробки проекту	Дає можливість отримувати нові дані для розробки
2	Висока вартість створення та забезпечення системи	Використання тільки мін. обороною	Неможливість виходу на інші ринки

У таблиці 4.8 показано ступеневий аналіз конкуренції на-ринку, а у таблиці 4.9 наведено аналіз конкуренції за М. Портером.

Таблиця 4.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополія/олігополія/ монополістична /чиста	Національний бізнес, воєнна сфера що є досить специфічною та закритою до розповсюдження	працює в рамках гос. програм
2. За рівнем конкурентної боротьби- локальний/національний/...	Міжнародний	Міноборони усіх країн роблять схожі сервіси для виявлення ворога

Продовження таблиці 4.8

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
3. За галузевою ознакою- міжгалузева/ внутрішньогалузева	Міжгалузева. Конкуренція на ринку ведеться між різними країнами	Необхідно зосередити зусилля на більш чіткий аналізу та влучній ідентифікації типів техніки

Продовження таблиці 4.8

5.3а характером конкурентних переваг - цінова / нецінова	Нецінова. При виборі методів та алгоритмів споживач(країна) дивиться на ефективність моделей та рівень якості ідентифікації системи. Вважаю що безпека країни та її воєнний потенціал та підготовка є першо степеню, тому країна не буду економити на розробці	Унікальність системи та її функціональність
6. 3а інтенсивністю - марочна/ не марочна	Марочна	Диференціація моделей та методів за метою задоволення потреб споживачів

Таблиця 4.9 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Процес	Висновки
Прямі конкуренти в галузі	Перелік прямих конкурентів	Прямих конкурентів на ринку України не визначено, або не існує, прогноз зробити неможливо, тому що ця технологія є воєнною розробкою.
Потенційні конкуренти	Визначити бар'єри входження в ринок	Бар'єри входу на ринок є великими, тому що треба заключити контракт з країною на розробку даної системи.
Постачальники	Визначити фактори сили постачальників	Дуже важливо отримувати багато чітких даних на яких модель може вчитись. Це є одним з ключових факторів у розробці даної системи.

Продовження таблиці 4.9

Складові аналізу	Процес	Висновки
Клієнти	Визначити фактори сили споживачів	Споживачами є країна, що робить осмислений вибір даного програмного рішення, це критична необхідність для розроблення подальшої стратегії чи наведення стратегічно важливої інформації
Товари замітники	Фактори загроз з боку заміників	Конкуренції зараз немає, чи ми не знаємо про її наявність через специфіку галузі розробки

Відповідно до даних вказаних вище, у сфері наразі немає конкуренції про яку ми можемо дізнатись явно.

Таким чином тип конкуренції на-ринку – монополістична конкуренція.

У таблиці 4.10 наведено пояснення та опис факторів конкурентоспроможності системи.

Таблиця 4.10 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування вибору
1	Частка ринку	Частка ринку є специфічною через специфіку клієнта розробки, який є монополістом у даній сфері. В таких умовах у нас є багато можливостей під час відсутності конкурентоспроможності у наших конкурентів.
2	Ціна	Споживач вложить стільки коштів скільки треба для отримання ідеального варіанту даного алгоритму
3	Асортимент	Зараз можливо є конкретність щодо споживача, але вона латентна. Споживач буде в першу чергу дивитись на новизну методів та чіткість ідентифікації моделі
4	Доступ до каналів розподілу	Маркетингу як такого немає. Доступ буде до цього софіта лише у споживача, тому що дана інформацію буде засекречена як мені здається.
5	Торговий маркетинг	
6	Рівень диференціації ТМ	В умовах відсутності конкуренції, важливо чітко презентувати свою ТМ, щоб показати споживачу унікальну цінність нашого алгоритму та нашої моделі у порівнянні із схожими технологіями

Продовження таблиці 4.10

8	Рівень лояльності до бренду	Відсутнє будь-яке розповсюдження.
9	Унікальність позиціонування	Важливим є створення та позиціювання даного продукту на фоні інших існуючих алгоритмів аналогів.
10	Маркетинговий бюджет	Маркетингу немає

4.3 Аналіз інноваційності проекту

Таблиця 4.11 показує аналіз сильних та слабких сторін проекту.

Таблиця 4.11 - Порівняльний аналіз сильних та слабких сторін

№ п/ п	Фактор конкурентоспроможності	Бал и 1- 20	Рейтинг товарів-конкурентів у порівнянні з нашою системою					
			- 3	- 2	- 1	0	+	+
1	Масштабованість	20					+	
2	Потреби споживачів	17				+		
3	Актуальність товару	16					+	
4	Підтримка та кастомізація продукту	18					+	
5	Підтримка і оновлення.	19				+		
6	Ціна та собівартість продукції	17	+					

Таблиця 4.12 - SWOT- аналіз інноваційного проекту

Сильні сторони: ● Ексклюзивність на ринку ● Зручність використання ● Оптимізований алгоритм	Слабкі сторони: ● Відсутність конкуренції ● Нестача належної кількості даних для навчання
--	---

У таблиці 4.13 наведені альтернативи ринкового впровадження інноваційного проекту. У 4.14 показано вибір цільових груп потенційних споживачів.

Таблиця 4.13 - Альтернативи ринкового впровадження інноваційного проекту

<i>№ п/п</i>	<i>Альтернатива (орієнтовний комплекс заходів) ринкової поведінки</i>	<i>Ймовірність отримання ресурсів</i>	<i>Строки реалізації</i>
1	Рекламна компанія спрямована на просування бренду	0%	немає
2	Впровадження програм лояльності та тестових періодів	100%	немає
3	Соціальні мережі «Facebook», «Telegram», «Instagram».	0%	немає
4	Тематичні заходи	0%	немає

Таблиця 4.14. - Вибір цільових груп потенційних споживачів

<i>№ п/п</i>	<i>Опис профілю цільової групи потенційних клієнтів</i>	<i>Готовність споживачів сприйняти продукт</i>	<i>Орієнтов ний попит в межах цільової групи (сегмент у)</i>	<i>Інтенсивність конкуренції в сегменті</i>	<i>Простота входу у сегмент</i>
1	Держава (мін. Оборона)	Висока – через потреби у даній технології у воєнний час	100%	Низька, в нашій країні не так багато подібних рішень чи немає зовсім	Легко, тому що немає аналогів

В таблиці 4.15 вказані базові стратегії розвитку, та у таблиці 4.16 базові стратегії конкурентної поведінки. У таблиці 4.17 – основні визначення стратегії позиціонування, та у 4.18 – ключові переваги концепції потенційного товару.

Таблиця 4.15 - Базові стратегії розвитку

<i>n/ n</i>	<i>Обрана альтернатива розвитку проекту</i>	<i>Стратегія охоплення ринку</i>	<i>Ключові конкурентоспромо жні позиції відповідно до обраної альтернативи</i>	<i>Базова стратегія розвитку*</i>
1	Впровадження інноваційних рішень на стартових етапах проекту. (Забезпечити високу точність ідентифікації та надійність роботи системи)	Недиференційований маркетинг	- Інноваційність	Стратегія технологічного лідера

Таблиця 4.16 - Визначення базової стратегії конкурентної поведінки

<i>No n/n</i>	<i>Чи є проект «першопрохідцем» на ринку?</i>	<i>Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?</i>	<i>Чи буде компанія копіювати основні характеристики товару конкурента, і які?</i>	<i>Стратегія конку- рентної поведінки*</i>
1	Так	Ні, компанія буде монополістом на ринку	Буде, якщо з'являться конкуренти з інтересними фічами конкуренти з інтересними фічами	Немає

Таблиця 4.17 - Основні визначення стратегії позиціонування

<i>№ n/ n</i>	<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія я розвитку</i>	<i>Ключові конкурентоспромо жні позиції власного стартап- проекту</i>	<i>Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)</i>
1	Оптимізація моделі для вирішення поставленої задачі	Стратегія диференціації	Швидка та точна ідентифікація	Швидко і точно, з можливістю навчання на нових датасетах

Таблиця 4.18 - Визначення ключових переваг концепції потенційного товару

<i>№ n/n</i>	<i>Потреба</i>	<i>Вигода, яку пропонує товар</i>	<i>Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)</i>
1	Ідентифікація воєнної техніки на знімках із супутника	Оперативна ідентифікації техніки та її типів на знімках із супутника	Відсутність конкурентів. А якби б були – оптимізація алгоритму та точність ідентифікації

Таблиця 4.19 – Опис трьох рівнів моделі товару

<i>Рівні товару</i>	<i>Сутність та складові</i>		
I. Товар за задумом	Система ідентифікації воєнної техніки на знімках із супутника, з ідентифікацією типів техніки		
II. Товар у реальному виконанні	– Властивості/характеристики – Апаратна платформа – Клієнт – Пристрої, що підтримуються – Можливість інтегрування з іншими системами	М/Нм М Нм М М	Вр/Тх Тл/Е/Ор Тх/Ор/Вр Тх/Тл/Ор Тх/Ор/Вр Тх/Ор/Вр Тх/Ор
III. Товар із підкріпленням	Якість: Повинні бути використані сучасні криптографічні стандарти та алгоритми (AES-192, TLS 1.2) Ідентифікація має бути коректною у 80-90% випадків Пакування: Електронна документація, що містить таку інформацію: ● Інструкція щодо користування. ● Необхідні технічні вимоги.		
	Марка: Ukraine Shield		

Таблиця 4.20 - Межі встановлення цін

<i>№ n/n</i>	<i>Рівень цін на товари- замінники</i>	<i>Рівень цін на товари- аналоги</i>	<i>Рівень доходів цільової групи споживачів</i>	<i>Верхня та нижня межі встановлення ціни на товар/послугу</i>
1	невідомо	немає	Юридичні особи > 500000 грн	200000 – 500000

В таблицях 4.21 та 4.22 вказані формування системи збуту, та основні концепції маркетингових комунікацій відповідно.

Таблиця 4.21 - Формування системи збуту

<i>No n/n</i>	<i>Специфіка закупівельної поведінки цільових клієнтів</i>	<i>Функції збуту, які має виконувати постачальник товару</i>	<i>Глибина каналу збуту</i>	<i>Оптималь на система збуту</i>
Підприємств а та організації	Платять за те, що може допомогти країні у воєнних цілях	Постачання та налаштування спеціалістами компанії.	Розробни к — клієнт	Офіційний портал з усією інформаці ю для клієнта на варіантами зв'язку.

Таблиця 4.23 - Основні концепції маркетингових комунікацій

<i>№ п/п</i>	<i>Специфіка поведінки цільових клієнтів</i>	<i>Канали комунікацій, якими користуються цільові клієнти</i>	<i>Ключові позиції, обрані для позиціонування</i>	<i>Завдання рекламного повідомлення</i>	<i>Концепція рекламного звернення</i>
Клієнт дізнається про продукт через приватні заходи	Приватні заходи	Інтернет, Професійні програмісти та аналітики	Точне, швидке, надійне рішення у сфері ідентифікації	немає	відсутня

ВИСНОВКИ ДО РОЗДІЛУ 4

У даному розділі було виконано документування й аналіз даного стартапу. Зазначено його сильні й слабкі сторони, що призвело до визначення технічних можливостей його створення.

Було зроблено висновок що на ринку немає конкуренції на даний момент. Стартап має технічні переваги за рахунок модифікованої моделі та більш точний алгоритм розпізнавання. Можлива поява великої конкуренції якщо хтось скопіює даний алгоритм.

На основі зазначеної стратегії позиціонування ми можемо зробити висновок, о необхідності зосередження на стратегії удосконалення та удосконалення існуючого алгоритму, його модифікації та розширення.

Проект являє собою цілісну систему для входження на ринок за попереднім оцінюванням. Цей проект принесе допомогу нашій країні у важкі часи, та допоможе у майбутньому. Допоможе нашим військовим аналітикам швидко та точно оцінювати загрозу, оцінювати пересування ворогу та розробляти контр дії.

ВИСНОВКИ

У ході виконання наукової роботи розроблена система ідентифікації воєнної техніки та її типів за знімком із супутника.

Оглянуті існуючі рішення у біометричній ідентифікації та комп'ютерного зору. Та з'ясовано, що найбільш придатним інструментом для вирішення поставленої задачі є машинне навчання.

Була обрана система ідентифікації на основі машинного навчання з учителем. Для цього був знайдений датасет, який використовувався для навчання. Також у цьому етапі були затверджені технології, що використовувались для розробки.

В ході роботи розроблені та протестовані системи навчання нейронних мереж, зібрані аналітичні дані та зроблені висновки. Вирішена проблема вибору алгоритму для детектування та ідентифікації об'єкту. Для ідентифікації створена модифікація для алгоритму згорткової нейронної мережі, що покращує розпізнавання за рахунок генеративно-змагальних мереж.

За отриманими даними виявлено, що багатошаровий перцептрон не відповідає задовільним результатам, та статистично відрізняє типи техніки, менш ніж з 20% вірогідністю. Наступним була навчена модель згорткової нейронної мережі, та завдяки її особливостям, правильність ідентифікації обраної моделі була доволі високою, біля 80-85%, що підтверджено шляхом тестування.

На основі ефективної згорткової нейронної мережі, був написаний модифікований алгоритм, що збільшив показники правильної ідентифікації до 90-95%. Це досягнуто завдяки розширенню даних та шарів відсіву, також збільшенню епох навчання. Була реалізована рання зупинка - припинення навчання моделі, якщо відстежувальні показники не покращуються протягом певної кількості епох. Встановлено, що важливо, щоб швидкість навчання була нижчою за швидкість ранньої зупинки, щоб модель мала можливість спробувати нову швидкість перед зупинкою.

Важливим аспектом вирішення даної задачі була наявність датасету, що відповідає всім вимогам нашої задачі. Дані були використані з Американських воєнних архівів 95-97-х років.

Таким чином найбільш ефективним є модифікований варіант реалізації згорткової нейронної мережі, із ранньою зупинкою, правильність якого сягаю 90-95%. Але робота з ідентифікації потребують подальших досліджень та модифікацій, особливо тестування на інших наборах даних та навчання моделей на нових датасетах.

ПЕРЕЛІК ПОСИЛАНЬ

- 1) Suhail A. CONVOLUTIONAL NEURAL NETWORK BASED OBJECT DETECTION: A REVIEW [Електронний ресурс] / A. Suhail, M. Jayabalan, V. Thiruchelvam // Advance Scientific Research. – 2020. – Режим доступу до ресурсу: <http://dx.doi.org/10.31838/jcr.07.11.140>.
- 2) Subcategory-Aware Convolutional Neural Networks for Object Proposals and Detection [Електронний ресурс] / Y.Xiang, C. Wongun, L. Yuanqing, S. Silvio // IEEE. – 2017. – Режим доступу до ресурсу: 10.1109/WACV.2017.108.
- 3) Agrawal P. Analyzing the Performance of Multilayer Neural Networks for Object Recognition [Електронний ресурс] / P. Agrawal, R. Girshick, J. Malik // ECCV. – 2014. – Режим доступу до ресурсу: https://link.springer.com/chapter/10.1007/978-3-319-10584-0_22.
- 4) Теорія використання tensorflow та keras.
URL: <https://www.tensorflow.org/tutorials/keras/classification>
- 5) Дані для датасету MSTAR
URL: <https://www.sdms.afrl.af.mil/index.php?collection=mstar>
- 6) Класифікація зображень за допомогою tensorflow.
URL: <https://www.tensorflow.org/tutorials/images/classification>
- 7) Орељен Жерон(2017). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems с. 99-112
- 8) A Beginner's guide to Deep Learning based Semantic Segmentation using Keras [Електронний ресурс] – Режим доступу: <https://divamgupta.com/image-segmentation/2019/06/06/deep-learning-semantic-segmentation-keras.html>
- 9) 9Image Segmentation Keras : Implementation of Segnet, FCN, UNet, PSPNet and other models in Keras. [Електронний ресурс] - Режим доступу: <https://github.com/divamgupta/image-segmentation-keras>
- 10) Y. Zoabi, S. Deri-Rozov, and N. Shomron, Machine learning-based

prediction of COVID-19 diagnosis based on symptoms, npj Digital Medicine, vol. 4, no. 1, Art. no. 1, Jan. 2021, doi: 10.1038/s41746-020-00372-6.

11) Прохоренко Николай та Дронов Владимир, Python 3. Самое необходимое, 2-е изд. БХВ-Петербург, 2019.

12) Теорія computer vision

URL: <https://www.ibm.com/topics/computer-vision>

ДОДАТКИ

ДОДАТОК А

ЛІСТИНГ ПРОГРАМНОГО КОДУ СИСТЕМИ

```

import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.preprocessing import image_dataset_from_directory
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import *
from tensorflow.keras.applications.vgg19 import VGG19
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix
import tensorflow_addons as tfa

params = {"ytick.color" : "w",
          "xtick.color" : "w",
          "axes.labelcolor" : "w",
          "axes.edgecolor" : "w",
          "figure.figsize" : (10,10),
          "axes.titlecolor" : 'w',
          "axes.facecolor" : 'w',
          "figure.facecolor" : 'k'}

colors = plt.rcParams['axes.prop_cycle'].by_key()['color']

image_size = (128,128)
batch_size = 32

train_ds = image_dataset_from_directory('mstar_imgs',
                                       subset='training',
                                       image_size=image_size,
                                       labels='inferred',
                                       validation_split=.2,
                                       seed=10,
                                       label_mode='categorical',
                                       color_mode='rgb',
                                       batch_size=batch_size)

val_ds = image_dataset_from_directory('mstar_imgs',

```

```

subset='validation',
image_size=image_size,
labels='inferred',
validation_split=.2,
seed=10,
label_mode='categorical',
color_mode='rgb',
batch_size=batch_size)

class_names = train_ds.class_names
with plt.rc_context(params):
    plt.figure(figsize=(10, 10))
    for images, labels in train_ds.take(1):
        for i in range(9):
            ax = plt.subplot(3, 3, i + 1)
            print(labels[i])
            plt.imshow(images[i].numpy().astype("uint8"), cmap='gray')
            plt.title(class_names[np.argmax(labels[i])], )
            plt.axis("off")

val_batches = tf.data.experimental.cardinality(val_ds)
test_ds = val_ds.take(val_batches // 5)
val_ds = val_ds.skip(val_batches // 5)

AUTOTUNE = tf.data.AUTOTUNE

train_ds = train_ds.cache().shuffle(1000).prefetch(buffer_size=AUTOTUNE)
val_ds = val_ds.cache().prefetch(buffer_size=AUTOTUNE)
test_ds = test_ds.cache().prefetch(buffer_size=AUTOTUNE)

num_classes=9

metrics = [
    keras.metrics.CategoricalAccuracy(name='categorical_accuracy'),
    tfa.metrics.MatthewsCorrelationCoefficient(num_classes=9, name='MCC'),
    tfa.metrics.FBetaScore(num_classes=9, average='weighted', beta=2.0,
name='F2'),
    keras.metrics.AUC(name='auc'),
    keras.metrics.AUC(name='prc', curve='PR'),
]

```

```

model_MLP = keras.Sequential()
model_MLP.add(InputLayer(input_shape=(image_size + (1,))))

model_MLP.add(Flatten())

model_MLP.add(Dense(100, activation='relu'))
model_MLP.add(Dense(100, activation='relu'))
model_MLP.add(Dense(100, activation='relu'))

model_MLP.add(Dense(num_classes, activation='softmax'))

model_MLP.summary()

model_MLP.compile(
    optimizer='adam',
    loss='categorical_crossentropy',
    metrics=metrics)

res_MPL = model_MLP.fit(train_ds, epochs=20, validation_data=val_ds)

def holdout_results(model):
    result = model.evaluate(test_ds)
    return dict(zip(model.metrics_names, result))

def plot_metrics(history):
    with plt.rc_context(params):
        metrics = ['loss', 'prc', 'F2', 'MCC']
        plt.figure(figsize=(10,10))
        for n, metric in enumerate(metrics):
            name = metric.replace("_", " ").capitalize()
            plt.subplot(2,2,n+1)
            plt.plot(history.epoch, history.history[metric], color=colors[0],
label='Train')
            plt.plot(history.epoch, history.history['val_'+metric],
color=colors[0], linestyle="--", label='Val')
            plt.xlabel('Epoch')
            plt.ylabel(name)
            if metric == 'loss':
                plt.ylim([0, plt.ylim()[1]])
            elif metric == 'auc':
                plt.ylim([0.8,1])

```

```

elif metric == 'MCC':
    plt.ylim([-1,1])
else:
    plt.ylim([0,1])

plt.legend();

def plot_cm(model, data):
    with plt.rc_context(params):
        y_true = []
        y_pred = []
        for x,y in data:
            y= tf.argmax(y,axis=1)
            y_true.append(y)
            y_pred.append(tf.argmax(model.predict(x),axis = 1))

        y_pred = tf.concat(y_pred, axis=0)
        y_true = tf.concat(y_true, axis=0)

        cm = confusion_matrix(y_true, y_pred)
        fig = plt.figure(figsize = (10,10))
        ax1 = fig.add_subplot(1,1,1)
        sns.set(font_scale=1.4) #for label size
        sns.heatmap(cm,cmap='binary',          annot=True,          fmt='d',
xticklabels=class_names, yticklabels=class_names, annot_kws={"size": 10},
                    cbar = False);
        ax1.set_ylabel('True Values',fontsize=14)
        ax1.set_xlabel('Predicted Values',fontsize=14)
        plt.show()

plot_metrics(res_MLP)

plot_cm(model_MLP, val_ds)

holdout_results(model_MLP)

plot_cm(model_MLP, test_ds)

model_CNN = tf.keras.Sequential([
    tf.keras.layers.InputLayer(input_shape=(image_size + (1,))),
    tf.keras.layers.Rescaling(1./255),

```



```

tf.keras.layers.Conv2D(32, 3, activation='relu'),
tf.keras.layers.MaxPooling2D(),
tf.keras.layers.Conv2D(32, 3, activation='relu'),
tf.keras.layers.MaxPooling2D(),
tf.keras.layers.Conv2D(32, 3, activation='relu'),
tf.keras.layers.MaxPooling2D(),
tf.keras.layers.Flatten(),
tf.keras.layers.Dense(128, activation='relu'),
tf.keras.layers.Dense(num_classes, activation='softmax')
])
model_CNN.summary()

model_CNN.compile(
    optimizer='adam',
    loss='categorical_crossentropy',
    metrics=metrics)

res_CNN = model_CNN.fit(train_ds, epochs=10, validation_data=val_ds)

plot_metrics(res_CNN)

plot_cm(model_CNN, val_ds)

holdout_results(model_CNN)

plot_cm(model_CNN, test_ds)

val_batches = tf.data.experimental.cardinality(val_ds)
test_ds = val_ds.take(val_batches // 5)
val_ds = val_ds.skip(val_batches // 5)

AUTOTUNE = tf.data.AUTOTUNE

train_ds = train_ds.cache().shuffle(1000).prefetch(buffer_size=AUTOTUNE)
val_ds = val_ds.cache().prefetch(buffer_size=AUTOTUNE)
test_ds = test_ds.cache().prefetch(buffer_size=AUTOTUNE)

callbacks=[
    keras.callbacks.EarlyStopping(patience=10, verbose=1,
monitor='val_F2', mode='max', restore_best_weights=True),
    keras.callbacks.ReduceLROnPlateau(factor=.5, patience=3, verbose=1)
]

```

```

    ]

my_model = tf.keras.Sequential([
    tf.keras.layers.InputLayer(input_shape=(image_size + (3,))),
    tf.keras.layers.Rescaling(1./255),
    tf.keras.layers.RandomFlip('horizontal_and_vertical'),
    tf.keras.layers.RandomRotation(0.2),
    tf.keras.layers.RandomZoom(0.1),
    tf.keras.layers.Conv2D(128, 5, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(64, 4, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(32, 3, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Conv2D(16, 2, activation='relu'),
    tf.keras.layers.MaxPooling2D(),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.Dense(256, activation='relu'),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(num_classes, activation='softmax')
])

my_model.summary()

my_model.compile(
    optimizer=keras.optimizers.Adam(1e-3),
    loss="categorical_crossentropy",
    metrics=metrics,)

last_res = my_model.fit(
    train_ds, epochs=100, callbacks=callbacks, validation_data=val_ds,
)

plot_cm(my_model, val_ds)

plot_metrics(last_res)
holdout_results(my_model)
plot_cm(my_model, test_ds)

```