

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційних радіотехнологій та технічного захисту інформації
(повна назва)

Кафедра мікропроцесорних технологій і систем
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

рівень вищої освіти другий (магістерський)
Система розпізнавання акустичного випромінювання дронів з використанням
нейронної мережі
(тема)

Виконав:
здобувач 2 року навчання,
групи ІМСм-23-1

Олег ШУНІБОРОВ

(власне ім'я, прізвище)

Спеціальність 171 Електроніка
(код і повна назва спеціальності)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Інженерія
мікропроцесорних систем
(повна назва освітньої програми)

Керівник проф. Олександр ВОРГУЛЬ
(посада, власне ім'я, прізвище)

Допускається до захисту

В.о. завідувача кафедри МТС



(підпис)

Олег ЗУБКОВ

(власне ім'я, прізвище)

2025 р.

5) Розробити схему електричну принципову та програмне забезпечення мікроконтролера

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

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

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата


КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Огляд та аналіз методів обробки та розпізнавання акустичних сигналів та аналогічних систем	21.04.2025-27.04.2025	
2	Аналіз спектральних властивостей акустичних випромінювань дронів та створення датасету	28.04.2025-7.05.2025	
3	Навчання та тестування нейронних мереж	8.05.2025-16.05.2025	
4	Розробка схеми електричної структурної пристроя розпізнавання акустичних випромінювань дронів	17.05.2025-24.05.2025	
5	Розробка принципової схеми та програмного забезпечення пристроя розпізнавання акустичних випромінювань дронів	25.05.2025-7.06.2025	
6	Написання пояснювальної записки	8.06.2025-15.06.2025	

Дата видачі завдання 21 04 _____ 2025р.

Здобувач _____ 

(підпис)

Керівник роботи _____  проф. Олександр ВОРГУЛЬ
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 83 с., 23 рис., 5 табл., 4 дод., 18 джерел.

ДРОН, АКУСТИЧНЕ ВИПРОМІНЮВАННЯ, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, КЕПСТРАЛЬНІ КОЕФІЦІЄНТИ, МІКРОКОНТРОЛЕР.

Об'єкт дослідження – задача реалізації розпізнавання акустичного випромінювання дронів на апаратній платформі з обмеженими ресурсами.

Предмет дослідження – розробка архітектури глибокої нейронної мережі для вбудування в апаратну платформу.

Мета роботи – проектування електронного пристрою та архітектури згорткової нейронної мережі для вбудування в електронний пристрій, навчання нейронної мережі та тестування її ефективності.

Методи дослідження – теоретичний (збір та структуризація теоретичного матеріалу і аудіозаписів для датасету), експериментальний (програмна реалізація нейронної мережі та її навчання). Методи розробки базуються на технологіях Python з фреймворком tensorflow.

У результаті роботи проведено теоретичний аналіз спектральних характеристик акустичного випромінювання дронів, за якими можна виконувати їх розпізнавання та архітектур нейронних мереж для розпізнавання. Виконано розробку датасету записів акустичних сигналів дронів, декількох архітектур згорткових нейронних мереж для аналізу MEL спектрограм, проведено навчання нейронних мереж та дослідження ефективності їх роботи на тестовому датасеті. Обрано архітектуру з мінімальними вимогами до вбудування на апаратній платформі з обмеженими ресурсами. Розроблено електричні структурну та принципову схеми електронного пристрою розпізнавання акустичного випромінювання дрону, алгоритм роботи мікроконтролера STM32MP157 та програмну реалізацію цього алгоритму.

ABSTRACT

Explanatory note: 83 pages, 23 figures, 5 tables, 34 appendices, 18 references.

DRONE, ACOUSTIC EMISSION, CONVOLUTIONAL NEURAL NETWORK, CEPSTRAL COEFFICIENTS, MICROCONTROLLER.

The object of research is the task of implementing drone acoustic emission recognition on a hardware platform with limited resources.

The subject of research is the development of a deep neural network architecture for integration into a hardware platform.

The aim of the work is the design of an electronic device and the architecture of a convolutional neural network for embedding into the electronic device, training the neural network, and testing its efficiency.

Research methods – theoretical (collection and structuring of theoretical materials and audio recordings for the dataset), experimental (software implementation of the neural network and its training). The development methods are based on Python technologies using the TensorFlow framework.

As a result of the work, a theoretical analysis of the spectral characteristics of drone acoustic emissions was carried out, based on which recognition can be performed, as well as an analysis of neural network architectures suitable for recognition tasks. A dataset of recorded drone acoustic signals was developed, several convolutional neural network architectures for MEL spectrogram analysis were designed, neural networks were trained, and their performance was tested on a validation dataset. An architecture was selected with minimal hardware requirements for integration into a resource-constrained hardware platform. The structural and schematic diagrams of an electronic device for drone acoustic emission recognition were developed, along with the STM32MP157 microcontroller operating algorithm and its software implementation.

ЗМІСТ

КАЛЕНДАРНИЙ ПЛАН	3
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 ОГЛЯД ТА АНАЛІЗ МЕТОДІВ ОБРОБКИ АКУСТИЧНИХ СИГНАЛІВ ТА АНАЛОГІЧНИХ СИСТЕМ.....	11
1.1 Особливості власного акустичного випромінювання БПЛА	11
1.2 Класифікація методів виявлення та розпізнавання корисних звуків на фоні акустичних шумів та інших звуків	12
1.3 Розрахунок MEL кепстральних коефіцієнтів	14
1.4 Метод головних компонент (РСА).....	16
1.5 Згорткові нейронні мережі	17
1.6 Опис LSTM нейронних мереж.....	23
1.7 Апаратні платформи для реалізації обробки акустичних сигналів та нейронних мереж	26
2 АНАЛІЗ СПЕКТРАЛЬНИХ ОСОБЛИВОСТЕЙ ВЛАСНИХ АКУСТИЧНИХ ВИПРОМІНЮВАНЬ ДРОНІВ ТА РОЗРОБКА ДАТАСЕТУ ДЛЯ НАВЧАННЯ НЕЙРОННИХ МЕРЕЖ.....	30
2.1 Ааналіз спектральних особливостей власних акустичних випромінювань дронів.....	30
2.2 Підготовка датасету акустичних випромінювань дронів для проведення навчання нейронних мереж.....	34
3 ПРОЕКТУВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ РОЗПІЗНАВАННЯ АКУСТИЧНИХ ВИПРОМІНЮВАНЬ ДРОНІВ ТА ЇХ НАВЧАННЯ.....	36
3.1 Проектування нейронних мереж	36
де:	37
3.2 Результати навчання нейронних мереж.....	43
3.3 Результати тестування нейронних мереж.....	46
4 РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ СТРУКТУРНОЇ.....	51

5 РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ ПРИНЦИПОВОЇ ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ МІКРОКОНТРОЛЕРА.....	54
5.1 Вибір мікрофону та мікрофонного підсилювача для акустичних вимірювань.....	54
5.2 Вибір керуючого мікроконтролеру	56
5.3 Вибір мікросхеми flash пам'яті	58
5.4 Вибір мікросхеми оперативної пам'яті.....	59
5.5 Вибір дисплею	59
5.6 Розробка алгоритму керуючої програми	60
5.7 Розробка програмного забезпечення.....	62
ВИСНОВОК	67
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	69
ДОДАТОК А Відомість кваліфікаційної роботи.....	73
Додаток Б	74
Додаток В.....	75
Додаток Г	76

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

CNN – згортова нейронна мережа

MFCC - Mel-Frequency Cepstral Coefficients

РСА – метод головних компонент

RNN – рекурентні нейронні мережі

STFT – короткочасне перетворення Фур'є

DCT – дискретне косинусне перетворення

АЦП – аналого-цифровий перетворювач

БПЛА – безпілотний літальний апарат

ВСТУП

Сучасний розвиток безпілотних літальних апаратів (БПЛА), зокрема квадрокоптерів, значно розширив можливості їх використання у цивільних та військових сферах. Однак, поряд із корисними застосуваннями, дрони також можуть становити потенційну загрозу безпеці, оскільки використовуються для нелегальної розвідки, контрабанди та навіть у військових конфліктах. Це створює нагальну потребу у розробці ефективних систем виявлення та класифікації таких об'єктів.

Традиційні методи детекції дронів, зокрема радіолокаційні та оптичні системи, мають низку обмежень: вони можуть неефективно працювати у складних погодних умовах або у міських середовищах з великою кількістю перешкод. У зв'язку з цим перспективним напрямом є використання методів акустичного аналізу, що дозволяє розпізнавати характерне звукове випромінювання дронів незалежно від погодних умов і освітленості.

Метою даної роботи є розробка системи розпізнавання акустичного випромінювання дронів на основі нейронної мережі. Для цього використано методи цифрової обробки сигналів та глибокого навчання, що забезпечують високу точність і швидкість розпізнавання.

Об'єкт дослідження в дипломній роботі – це процес розпізнавання акустичного випромінювання безпілотних літальних апаратів (БПЛА). Предмет дослідження – методи та алгоритми обробки акустичних сигналів дронів з використанням нейронних мереж для їх виявлення та класифікації.

У роботі розглянуто основні принципи роботи нейронних мереж для обробки звукових сигналів, досліджено методи виділення акустичних характеристик дронів, а також реалізовано та протестовано модель глибокого навчання для їх ідентифікації. Запропонована система може знайти

застосування у системах безпеки, протидії несанкціонованим польотам та моніторингу повітряного простору.

1 ОГЛЯД ТА АНАЛІЗ МЕТОДІВ ОБРОБКИ АКУСТИЧНИХ СИГНАЛІВ ТА АНАЛОГІЧНИХ СИСТЕМ

1.1 Особливості власного акустичного випромінювання БПЛА

Для побудови методів виявлення та розпізнавання БПЛА за їх власним акустичним випромінюванням необхідно проаналізувати особливості цього випромінювання у часовій та частотній областях. Акустичне випромінювання БПЛА має специфічні характеристики як у часовій, так і в спектральній областях, що дозволяє відрізнити їх від інших джерел шуму.

У часовій області акустичні сигнали БПЛА демонструють періодичні коливання, пов'язані з обертанням пропелерів та роботою двигунів. Це проявляється у регулярних пульсаціях амплітуди сигналу. Також в акустичних сигналах БПЛА спостерігаються імпульсні складові, зумовлені турбулентними потоками повітря та взаємодією конструктивних елементів дрона з навколишнім середовищем [1,2,3].

У спектрі акустичного випромінювання БПЛА присутні чітко виражені основні частоти, що відповідають обертам двигунів, а також їх вищі гармоніки. Акустичне випромінювання БПЛА також характеризується широкосмуговим спектром, що охоплює частоти від десятків герць до 8-10кГц [1,2].

Ці особливості в часовій та спектральній областях є ключовими для розробки ефективних методів виявлення та розпізнавання БПЛА на основі аналізу їх акустичних сигнатур.

1.2 Класифікація методів виявлення та розпізнавання корисних звуків на фоні акустичних шумів та інших звуків

В діапазоні акустичних хвиль до 20кГц існує великий рівень шуму та завад, що формується голосами людей, двигунами автомобілів і т.і. У таблиці 1 можна побачити рівні шуму, що створюють різні джерела [4]

Таблиця 1 – Рівні шуму різних джерел

Noise Source	dB Level					
	Average		Minimum		Maximum	
	ALX	ZAG	ALX	ZAG	ALX	ZAG
cars/taxis	70.8	67.6	55.9	31.3	<u>94.6</u>	<u>92.0</u>
bikes/motorcycles	70.2	66.1	61.2	38.0	85.3	89.2
public transport	73.4	-	60.9	-	91.1	-
street vendors	72.0	66.9	55.7	53.7	94.3	82.5
social gatherings	71.8	65.7	61.7	50.0	<u>94.6</u>	90.5
trams	69.9	68.8	63.4	55.4	<u>77.2</u>	88.0
DJs/loudspeakers	70.7	63.7	58.1	54.9	87.1	79.4
industrial machines	-	<u>80.1</u>	-	<u>64.7</u>	-	90.2
construction work	73.6	70.1	<u>69.9</u>	52.7	78.1	90.6
commercial activities of shops	69.9	66.8	54.0	53.2	<u>94.6</u>	85.6
large trucks	-	67.3	-	42.5	-	89.3
loading and unloading	72.6	67.0	66.1	54.9	91.5	84.3
coffee shops	70.2	64.3	57.6	52.4	84.6	91.3
air conditioners	<u>73.9</u>	-	68.1	-	79.4	-
street performers	-	70.9	-	60.4	-	80.8
street fights	66.3	-	61.4	-	74.5	-
parking lots	71.5	62.8	58.5	31.3	91.5	88.3
bus stations	73.0	59.7	65.1	55.1	81.2	68.4

Одним із найголовніших питань при виявленні БПЛА є максимізація відстані, на якій можливе виявлення, а чим більше відстань, тим менший корисний сигнал на вході приймального пристрою і тим нижче відношення корисний сигнал/завада, що особливо актуально для міст, трас і т.д.. Тому

завдання виділення і розпізнавання власного сигналу акустичного випромінювання дрона є складною і потребує наукових досліджень.

Існує кілька основних методів виявлення та розпізнавання корисних звуків на фоні акустичних шумів та інших звуків. Вони поділяються на класичні методи обробки сигналів і методи, що використовують штучний інтелект та нейронні мережі.

До класичних методів обробки сигналів відносяться:

- Використання смугових фільтрів для виділення частот, характерних для цільового сигналу.
- Адаптивна фільтрація, яка динамічно налаштовує характеристики фільтра залежно від умов шуму.
- Вейвлет-перетворення дозволяє виділити корисні компоненти сигналу, зберігаючи їх часові характеристики.
- Короткочасне перетворення Фур'є (STFT) застосовується для аналізу частотного складу сигналу у певні моменти часу.
- Спектральне віднімання дозволяє оцінити і зменшити рівень шуму, віднімаючи спектр фонових звуків.
- Методи статистичного моделювання шуму, такі як Wiener-фільтр, ефективно пригнічують завади.

До методів штучного інтелекту та машинного навчання відносяться:

- Глибокі згорткові нейронні мережі (CNN) ефективно розпізнають особливості спектрограм цільових звуків.
- Рекурентні нейронні мережі (RNN) та їх модифікація LSTM добре підходять для аналізу часових послідовностей.
- Варіаційні автоенкодера (VAE) можуть бути використані для створення моделей фонових шумів і їх усунення із записаного сигналу.

- Generative Adversarial Networks (GANs) навчаються відрізняти корисний сигнал від шуму та можуть використовуватися для відновлення слабких акустичних сигналів.
- Метод незалежного компонентного аналізу (ICA) дозволяє розділити змішані акустичні сигнали на окремі джерела.
- Non-negative Matrix Factorization (NMF) ефективний у випадках, коли потрібно виділити певні частотні компоненти, характерні для корисного сигналу.

Комбіновані підходи

Останнім часом ефективним є поєднання класичних методів та глибокого навчання. Наприклад, STFT або вейвлет-перетворення можуть використовуватися для попередньої обробки, а потім отримані спектрограми подаються на вхід нейронної мережі для класифікації чи розпізнавання.

Розглянемо більш детально деякі з цих методів

1.3 Розрахунок MEL кепстральних коефіцієнтів

Mel-Frequency Cepstral Coefficients (MFCC) є широко використовуваними ознаками в задачах розпізнавання акустичних сигналів, зокрема для ідентифікації джерел звуку [5,6]. Даний метод заснований на особливостях людського слуху, який сприймає звуки нелінійно у частотній області. У цьому розділі розглянемо основний алгоритм обчислення MFCC та відповідні математичні формули.

Алгоритм обчислення MFCC складається з таких основних етапів:

1). Розбиття сигналу на вікна. Оскільки акустичний сигнал є нестационарним, його аналіз проводиться у короткочасних інтервалах. Вхідний сигнал $x(n)$

розбивається на короткі сегменти довжиною N із перекриттям P , що зазвичай становить 25–50% від довжини вікна. Для згладжування країв застосовується віконна функція Ганна:

$$w(n) = 0.5 \left(1 - \cos \frac{2\pi n}{N-1} \right), 0 \leq n \leq N \quad (1.1)$$

2) Обчислення спектра сигналу. Для кожного вікна сигналу виконується перетворення Фур'є з використанням швидкого алгоритму (FFT):

$$X(k) = \sum_{n=0}^{N-1} x(n)w(n)e^{-j2\pi kn/N}, k = 0, 1, \dots, N-1 \quad (1.2)$$

Далі визначається потужність спектра:

$$P(k) = |X(k)|^2 \quad (1.3)$$

3) Перетворення спектра у шкалу Mel. Людське вухо нелінійно сприймає частоти, тому використовується шкала Mel, що визначається як:

$$f_{mel} = 2595 \cdot \log_{10} \left(1 + \frac{f}{700} \right) \quad (1.4)$$

Задається набір трикутних фільтрів, рівномірно розташованих у шкалі Mel. Вихід кожного фільтра M обчислюється як зважена сума потужності спектральних складових:

$$S_m = \sum_{k=f_{m-1}}^{f_{m+1}} P(k)H_m(k) \quad (1.5)$$

де $H_m(k)$ — коефіцієнти трикутного фільтра, а f_{m-1} і f_{m+1} — межі фільтра.

4) Обчислення кепстрального коефіцієнта. До отриманих значень S_m застосовується дискретне косинусне перетворення (DCT), що дозволяє зменшити кореляцію між коефіцієнтами:

$$C_n = \sum_{m=1}^M \log(S_m) \cos \left[\frac{\pi n(m-0.5)}{M} \right], n = 1, 2, \dots, L, \quad (1.6)$$

де L — кількість коефіцієнтів MFCC, що зазвичай вибирається у межах 12–20.

5) Динамічні ознаки. Для покращення якості розпізнавання звуків обчислюють перші та другі похідні (дельта-коефіцієнти):

$$\Delta C_n = \frac{\sum_{i=1}^K i(C_{n+i} - C_{n-i})}{2 \sum_{i=1}^K i^2} \quad (1.7)$$

Зазвичай вибирають $K=2$, що забезпечує згладжування похідних.

MFCC є ефективним методом виділення ознак акустичного сигналу завдяки використанню шкали Mel та дискретного косинусного перетворення. Отримані коефіцієнти використовуються для класифікації та розпізнавання звуків, зокрема в задачах детекції БПЛА.

1.4 Метод головних компонент (PCA)

Метод головних компонент (PCA) використовується для зменшення розмірності вхідних ознак та підвищення ефективності класифікації [7,8]. PCA дозволяє визначити головні напрямки варіації даних і відкинути менш значущі компоненти.

Алгоритм обчислення PCA складається з наступних кроків:

1. Формується матриця ознак X розміром $N \times D$, де N — кількість вибірок, D — кількість ознак.
2. Обчислюється середнє значення кожної ознаки та віднімається від матриці X .
3. Обчислюється коваріаційна матриця:

$$C = \frac{1}{N} X^T X \quad (1.8)$$

4. Виконується власне розкладання коваріаційної матриці:

$$C \cdot v_i = \lambda_i v_i \quad (1.9)$$

де v_i — власні вектори, λ_i — власні значення.

5. Відбираються k головних компонент із найбільшими λ_i , що містять основну інформацію про дані.
6. Матриця ознак проектується на обрані головні компоненти:

$$X' = XV_k \quad (1.10)$$

де V_k — матриця головних компонент.

РСА допомагає зменшити розмірність MFCC-кепстральних коефіцієнтів, зменшити шум та прискорити навчання нейронних мереж.

1.5 Згорткові нейронні мережі

Згорткова нейронна мережа (Convolutional Neural Network, CNN) є однією з найпотужніших архітектур для обробки даних, що мають просторову або часову структуру [9-12]. Вона широко використовується для розпізнавання зображень, обробки звуку та відео. Основна ідея CNN полягає у виділенні локальних особливостей сигналу за допомогою згорткових фільтрів та поступовому зменшенні розмірності для отримання високорівневих ознак. В основному згорткові нейронні мережі використовуються для обробки зображень. Приклад згорткової мережі показаний на рисунку 1.1.

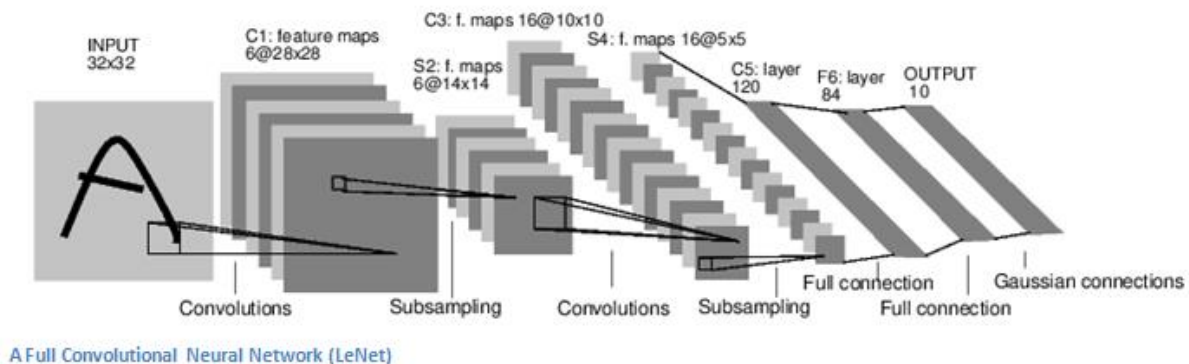


Рисунок 1.1 – Приклад структури загорткової нейронної мережі

З рисунку 1.1 ми можемо побачити, що згорткова нейронна мережа складається з наступних шарів

1. Згортковий шар (Convolution Layer)
2. Шар підвибірки (Subsampling Layer)
3. Повнозв'язний шар (Fully Connected Layer)

На вхід мережі (згортків шар) подається зображення. Розмір зображення зазвичай може бути від 28x28 пікселів до 640x640.

Згортковий шар є основним елементом CNN. Він застосовує фільтри (ядра згортки), які сканують вхідні дані та виділяють локальні особливості. Операція згортки описується формулою:

$$h_{ij} = \sum_m \sum_n x_{(i+m)(j+n)} w_{mn} + b \quad (1.11)$$

де:

- x — вхідний сигнал,
- w — ядро згортки,
- b — зміщення (bias),
- h_{ij} — вихідне значення згортки.

Цей процес дозволяє мережі виявляти такі особливості, як частотні піки, гармоніки або характерні патерни звуку. Візуально принцип згортки показаний на рисунку 1.2

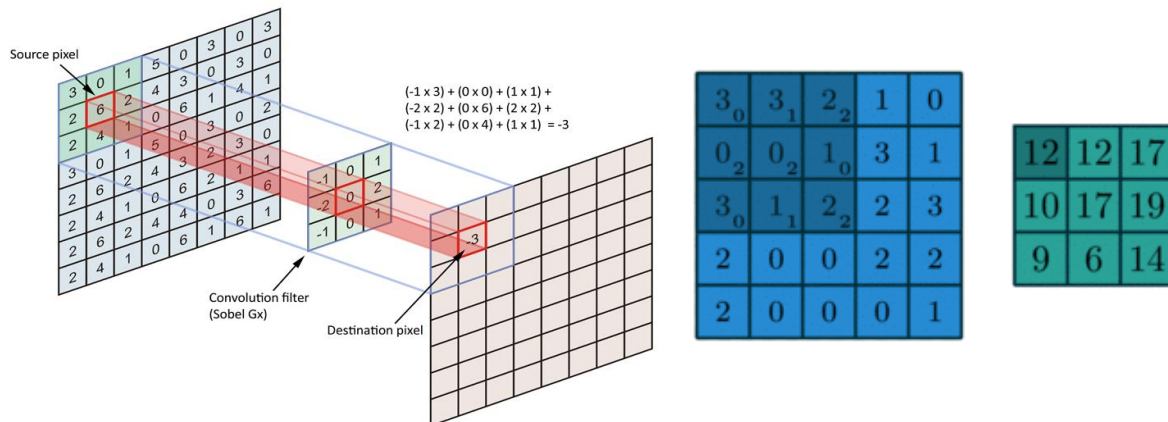


Рисунок 1.2 – Приклад згортки зображення з ядром

Після операції згортки маємо так звані карти ознак. Кожна карта є результатом згортки з відповідним ядром. Кожне ядро призначене для виділення своїх особливостей зображення: вертикальних чи горизонтальних меж, кутів і т.д. Кількість ядер зазвичай обирають від 16 до 64. Результат згортки обов'язково проходить через функцію активації. Найбільш популярними функціями активації є:

- 1) Лінійна. Виходи мережі є лінійними комбінаціями входів (Рисунок 3.2а).
- 2) Порогова. Ця функція використовувалася в оригінальній моделі штучного нейрону. Має центрований аналог ($\text{sign } x$), не диференційована (Рисунок 3.2б).
- 3) Сігмоїда.

$$f(x) = \frac{1}{1 + e^{-\alpha x}} \quad (1.12)$$

Довгий час вважалася функцією, яка найкраще описує роботу нейрона. Диференційована. Має поріг насичення (Рисунок 3.2в).

- 4) Tanh.

$$f(x) = \tanh\left(\frac{\alpha x}{2}\right) = \frac{1 - e^{-\alpha x}}{1 + e^{-\alpha x}} \quad (1.13)$$

Диференційована. Центрована. Має поріг насичення (Рисунок 3.2г).

- 5) ReLU(rectified linear unit).

$$f(x) = \max(0, x) \quad (1.14)$$

Візуальний вигляд функцій активації наведено на рисунку 1.3

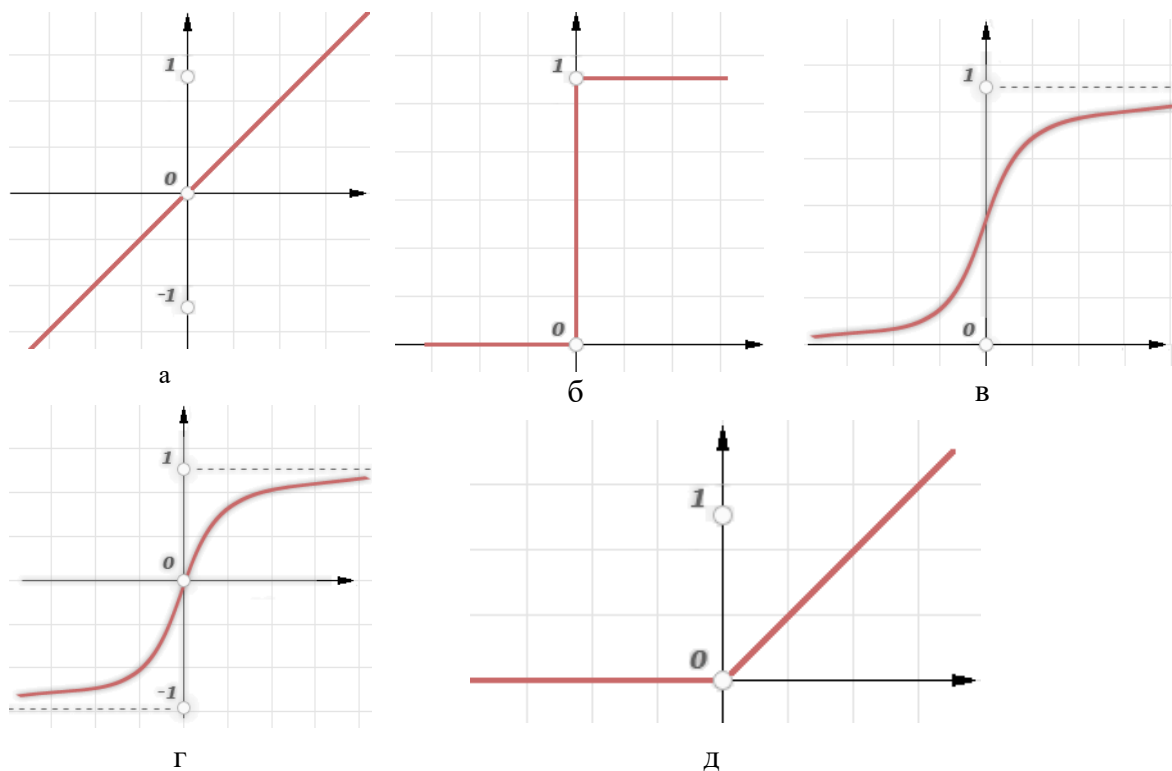


Рисунок 1.3 – Найбільш популярні функції активації (а – лінійна, б – порогова, в – сигмоїда, г – Tanh, д - ReLU)

Найчастіше застосовується функція ReLU. ReLU дозволяє нейронній мережі ефективно навчатися складним залежностям, усуваючи негативні значення та прискорюючи процес оптимізації.

Дослідниками нейронних мереж було доказано, що отримані карти ознак мають надлишковий розмір. Його можна зменшити у 2 рази по вертикалі та 2 рази по горизонталі, фактично не зменшивши кількість корисної інформації у картах ознак. Тому у сучасні нейронні мережі вводять шари підвибірки. Шар підвибірки (Pooling) зменшує розмірність ознак та підвищує стійкість до шумів. Найпопулярнішим методом є максимальна підвибірка (Max Pooling):

$$p_{ij} = \max_{m,n} h_{(i+m)(j+n)} \quad (1.15)$$

де обирається максимальне значення у вікні фіксованого розміру.

Принцип роботи цього шару показаний на рисунку 1.4

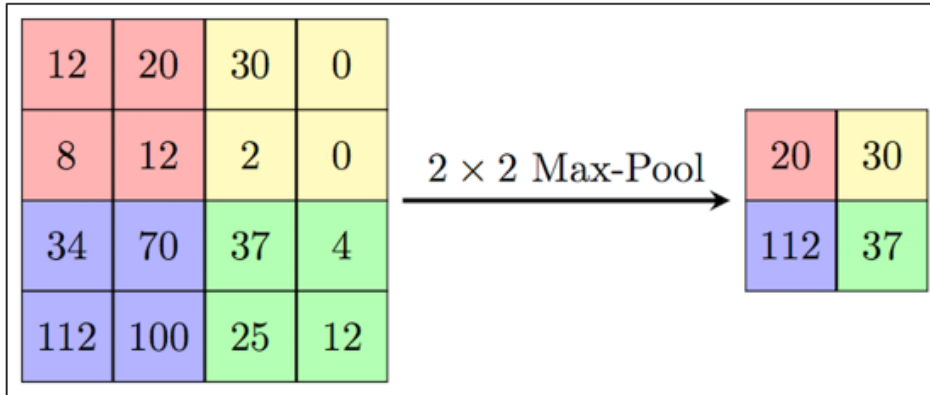


Рисунок 1.4 – Принцип роботи шару MaxPooling

Усе зображення розбивається на квадрати 4x4 пікселя. В кожному з таких квадратів є 4 менших квадрата 2x2. Для кожного з них знаходять найбільше значення, яке записується як результат операції Max-Pool у вихідну комірку матриці 2x2. Результат операцій згортки з різними ядрами, дію функції активації та операцію Max-Pool можна побачити візуально на рисунку 1.5

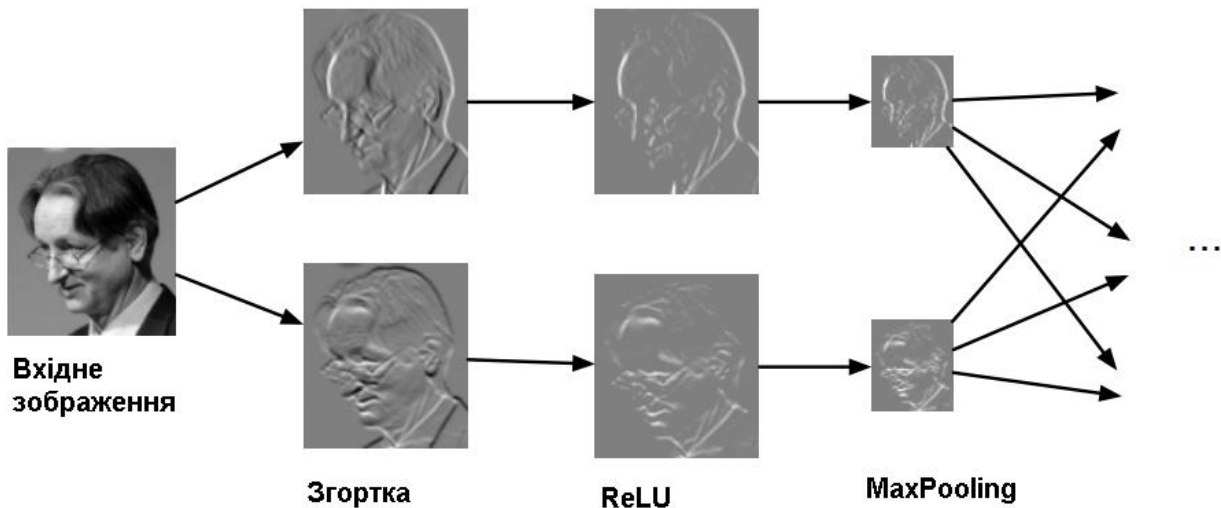


Рисунок 1.5 – Результат роботи згортки, функції активації, операції Max-Pool

Однак згорткові шари не дають можливість класифікувати об'єкти. Тому після кількох згорткових і підвибіркових шарів дані передаються до повнозв'язних шарів. Загальний вигляд такої мережі наведено на рисунку 1.6.

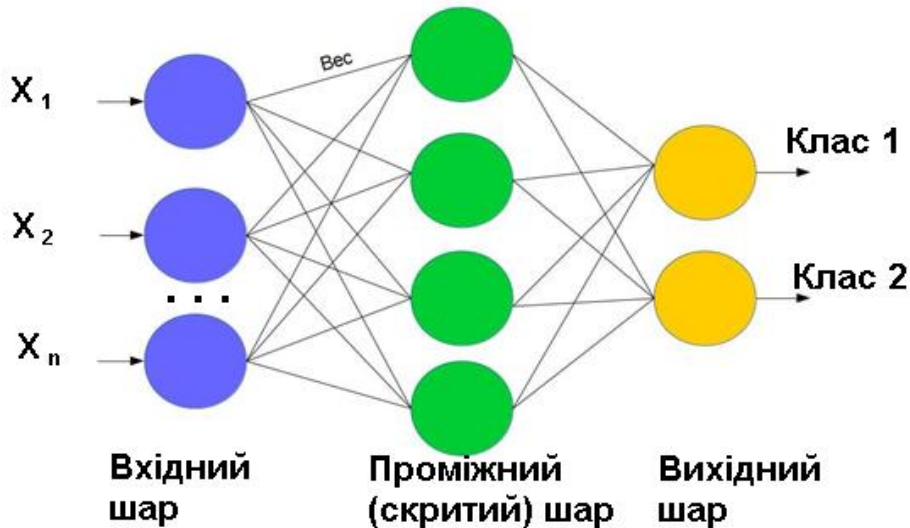


Рисунок 1.6 – Загальний вигляд нейронної мережі із повнозв'язних шарів

Він відповідає за остаточну класифікацію або регресію. Вихідне значення сигналу на кожному з нейронів визначається як:

$$y = \sigma(Wh + b)$$

де:

- W — матриця ваг,
- h — вектор ознак із попереднього шару,
- b — зміщення,
- σ — функція активації (наприклад, softmax для класифікації).

Для запобігання перенавчанню використовується нормалізація пакетів (Batch Normalization) та метод відсіву нейронів (Dropout). Це дозволяє покращити узагальнюючу здатність моделі та стабілізувати процес навчання. Згортова нейронна мережа також є потужним інструментом для обробки акустичних сигналів. Завдяки багатосаровій структурі та ефективній обробці локальних особливостей, CNN демонструє високу точність у розпізнаванні звуків, зокрема для ідентифікації акустичних сигналів дронів.

1.6 Опис LSTM нейронних мереж

Long Short-Term Memory (LSTM) – це тип рекурентної нейронної мережі (RNN), який ефективно працює із послідовними даними [13]. LSTM вирішує проблему зникання градієнта, яка притаманна класичним RNN, завдяки спеціальній структурі комірки пам'яті (memory cell).

LSTM складається з трьох основних елементів керування інформаційними потоками:

1. Вхідний шлюз (input gate) – контролює, яка нова інформація додається до стану пам'яті.
2. Шлюз забування (forget gate) – вирішує, яка інформація повинна бути забута.
3. Вихідний шлюз (output gate) – визначає, яка інформація передається до наступного етапу.

Ці механізми дозволяють LSTM зберігати важливу інформацію протягом тривалого часу та ігнорувати незначні дані.

Переваги LSTM для розпізнавання акустичних сигналів – це:

- Здатність працювати з послідовними даними: Акустичні сигнали є часовими рядами, і LSTM може ефективно аналізувати їхню динаміку та закономірності.
- Пам'ять на довгих інтервалах – Завдяки механізму збереження пам'яті, LSTM може фіксувати довгострокові залежності в акустичних сигналах.
- Устойчивість до шуму – модель може фільтрувати незначні коливання та фоновий шум.
- Гнучкість в обробці різних частотних компонентів – LSTM ефективно аналізує спектральні та тимчасові характеристики сигналу.

Недоліками архітектури LSTM є:

- Високі обчислювальні витрати: LSTM вимагає значних ресурсів для навчання та інференсу, особливо на великих наборах даних.
- Проблеми з надмірною пам'яттю: У разі дуже тривалих послідовностей модель може перенасичуватись інформацією.
- Необхідність ретельного налаштування гіперпараметрів: Параметри, такі як кількість шарів, кількість нейронів та швидкість навчання, значно впливають на якість розпізнавання.

Структура нейронної мережі LSTM наведена на рисунку 1.7

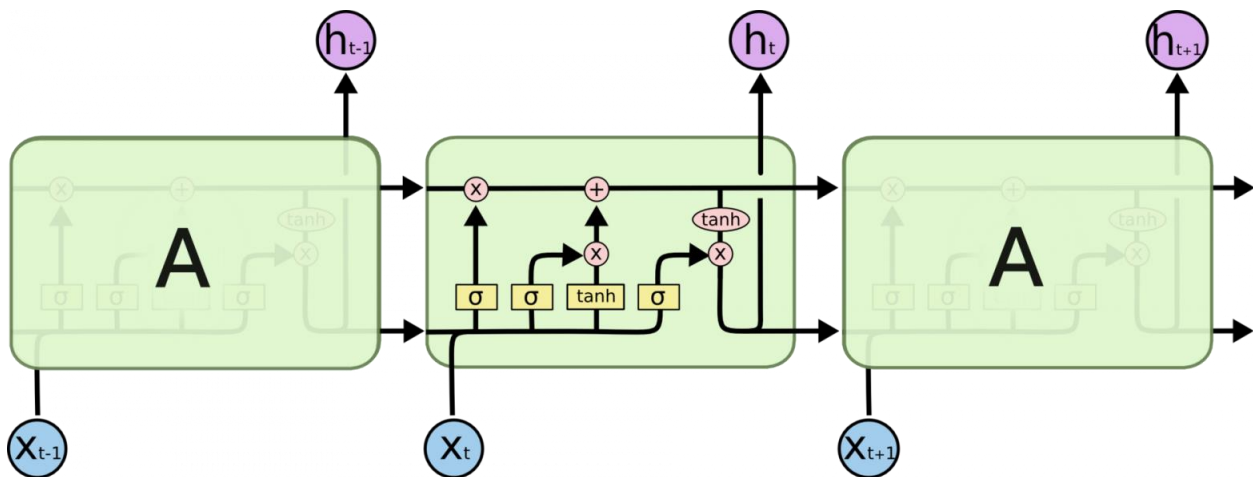


Рисунок 1.7 – Архітектура нейронної мережі LSTM

Перший крок у LSTM – визначити, яку інформацію можна викинути зі стану комірки. Це рішення приймає сигмоїдальний шар, званий шаром фільтра забування (forget gate layer). Він дивиться на i і повертає число від 0 до 1 для кожного числа зі стану комірки. 1 означає "повністю зберегти", а 0 - "повністю викинути". Наступний крок – вирішити, яка нова інформація зберігатиметься у стані осередку. Цей етап складається із двох частин. Спочатку сигмоїдальний шар під назвою шар вхідного фільтра (input layer gate) визначає, які значення слід оновити. Потім tanh-шар будує вектор нових значень-кандидатів, які можна додати до осередку. Зрештою, потрібно вирішити, яку інформацію ми хочемо отримувати на виході. Вихідні дані будуть засновані на нашому стані осередку, до них будуть застосовані деякі фільтри. Спочатку ми застосовуємо сигмоїдальний шар, який вирішує, яку інформацію зі стану осередку ми виводитимемо. Потім значення стану осередку проходять через tanh-шар, щоб отримати на виході значення з діапазону від -1 до 1, і перемножуються з вихідними значеннями шару сигмоїдального, що дозволяє виводити тільки необхідну інформацію.

При використанні архітектури LSTM для розпізнавання акустичних сигналів застосовується наступний алгоритм:

1. Вхідний шар отримує спектрограму або MFCC (Mel-frequency cepstral coefficients) з акустичного сигналу.
2. Шар LSTM вивчає часові залежності в сигналі.
3. Повнозв'язний шар (Dense) класифікує дані за відповідними категоріями (фон, дрон, вертоліт).
4. Функція активації softmax визначає ймовірності для кожного класу.

Порівняння цієї мережі з іншими моделями наведено у таблиці 1.2

Таблиця 1.2 – Порівняльний аналіз нейронної мережі LSTM

Параметр	LSTM	CNN	GRU
Пам'ять на довгих інтервалах	Висока	Низька	Середня
Обчислювальні витрати	Висока	Низька	Середня
Стійкість до шуму	Висока	Низька	Середня

LSTM є потужним інструментом для обробки акустичних сигналів завдяки здатності працювати з часовими рядами та враховувати довгострокові залежності. Однак, висока складність та обчислювальні витрати роблять цю мережу більш підходящою для застосувань з доступом до потужних апаратних ресурсів, таких як GPU або FPGA.

1.7 Апаратні платформи для реалізації обробки акустичних сигналів та нейронних мереж

Сучасні апаратні платформи для зчитування акустичних сигналів та обробки сигналів і зображень за допомогою нейронних мереж охоплюють широкий спектр пристроїв, таких як мікроконтролери, графічні процесори (GPU), програмовані логічні інтегральні схеми (FPGA) та спеціалізовані нейроморфні чипи. Кожна з цих платформ має свої переваги та обмеження, що визначають їхню ефективність у конкретних застосуваннях.

Використання мікроконтролерів (MCU), таких як Arduino, STM32, ESP32 має наступні переваги:

- низьке енергоспоживання, що важливо для вбудованих систем і пристроїв IoT;

- простота інтеграції з датчиками та периферійними пристроями;
- вартість та компактність.

Недоліки таких платформ:

- Обмежена обчислювальна потужність.
- Мала кількість оперативної пам'яті для роботи з великими нейронними мережами.
- Низька швидкість обробки даних у порівнянні з GPU та FPGA.

Використання графічних процесорів (GPU), таких як NVIDIA RTX 3080, NVIDIA Jetson Nano, AMD Radeon Instinct має наступні переваги:

- Висока продуктивність для паралельної обробки великих обсягів даних.
- Оптимізація для нейронних мереж та роботи з великими масивами зображень та аудіосигналів.
- Широка підтримка фреймворків машинного навчання (TensorFlow, PyTorch).

Їх недоліки:

- Високе енергоспоживання.
- Висока вартість обладнання.
- Потреба у додатковому охолодженні для ефективної роботи.

Також популярними є програмовані логічні інтегральні схеми (FPGA), такі як: Xilinx Zynq, Intel Stratix, Altera Cyclone. Їх перевагами є:

- Можливість гнучкої налаштування архітектури для конкретних завдань.
- Висока ефективність у реальному часі.
- Низьке енергоспоживання порівняно з GPU.

Недоліки цих платформ:

- Складність програмування та налаштування.

- Висока вартість розробки та відлагодження.
- Обмежені ресурси для роботи з великими моделями нейронних мереж.

Найновітнішою апаратною платформою є нейроморфні чипи: Intel Loihi, IBM TrueNorth, BrainChip Akida. Вони мають значні переваги:

- Архітектура, що імітує біологічні нейронні мережі.
- Енергоефективність та висока швидкість обробки сигналів.
- Підтримка обробки подій в реальному часі (event-based processing).

Їх недоліками є:

- Висока вартість та обмежена доступність.
- Обмежена сумісність з традиційними фреймворками.
- Потреба у спеціалізованому програмному забезпеченні.

Вибір апаратної платформи для зчитування акустичних сигналів та обробки сигналів і зображень залежить від конкретних вимог до продуктивності, енергоспоживання та вартості. Мікроконтролери підходять для простих завдань з низьким енергоспоживанням, тоді як GPU забезпечують високу продуктивність для навчання складних нейронних мереж. FPGA пропонують гнучкість і ефективність для задач реального часу, а нейроморфні чипи відкривають нові горизонти у створенні біологічно натхнених систем обробки даних.

Висновки. Аналіз наукових публікацій з виявлення акустичного випромінювання дронів показав, що вони створюють специфічні шуми при обертанні пропелерів. Такі сигнали мають характерні особливості у спектральній області, що використовуються для знаходження та розпізнавання таких акустичних випромінювань. Найбільш популярними методами виявлення спектральних особливостей є метод MEL кепстральних

коефіцієнтів та метод головних компонент. При цьому перший метод широко використовується разом із згортковими мережами при розпізнаванні акустичних сигналів. Проведений аналіз сучасних нейронних мереж показав, що на початку складної нейронної мережі слід використовувати декілька загорткових шарів, а після них необхідні повнозв'язні шари або рекурентна мережа LSTM. Практична реалізація нейронної мережі можлива на багатьох сучасних апаратних платформах. Але їх вибір суттєво залежить від складності розробленої нейронної мережі та обмежень на вартість і може бути обрана тільки по завершенні наукових досліджень з синтезу та тестування нейронної мережі.

2 АНАЛІЗ СПЕКТРАЛЬНИХ ОСОБЛИВОСТЕЙ ВЛАСНИХ АКУСТИЧНИХ ВИПРОМІНЮВАНЬ ДРОНІВ ТА РОЗРОБКА ДАТАСЕТУ ДЛЯ НАВЧАННЯ НЕЙРОННИХ МЕРЕЖ

2.1 Аналіз спектральних особливостей власних акустичних випромінювань дронів

Для проведення досліджень необхідно було мати сукупність записів акустичних сигналів власних випромінювань різних модифікацій дронів. Такі записи дозволяють оцінити спектральні особливості цих акустичних сигналів та провести навчання нейронних мереж.

В експериментах були задіяні дві моделі дронів: DJI Phantom 3 та Syma X5SW. Для запису звуків був обраний конденсаторний мікрофон Superlux ECM-999. Для запису аудіо використовувалась зовнішня звукова карта U-Phoria UM2. Записи аудіо були зроблені з частотою дискретизації 44100Гц у форматі wav файлів. Також були зроблені записи звуків транспорту, потягів, мовлення людей, птахів, шуму вітру. Основною особливістю зроблених аудіо записів польотів дронів є фіксація відстані між мікрофоном та дроном, що дає в подальшому можливість оцінити якість розпізнавання на різних відстанях.

Використовуючи отримані записи власного звукового випромінювання при польоту дрона на різних відстанях були побудовані спектральні щільності потужності (СЩП) сигналів з використанням швидкого перетворення Фур'є. Вони представлені на рисунку 2.1

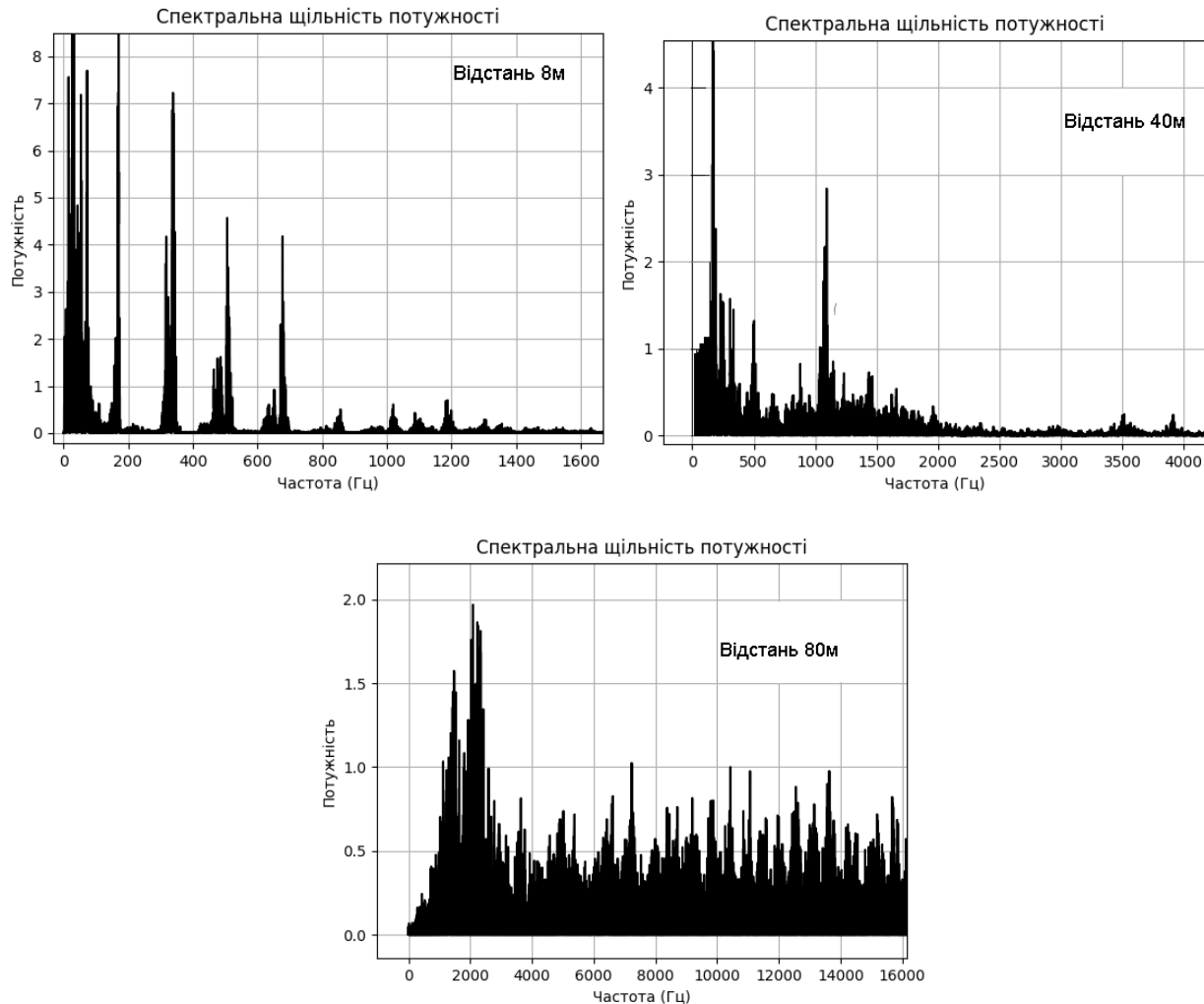


Рисунок 2.1 – Приклади спектральної щільності потужності власного акустичного випромінювання дронів на різних відстанях

З отриманих спектрів на рисунку 2.1 можна побачити, що спектральна щільність потужності має найпотужніший пік на частоті обертання двигунів. Також є багато гармонік основної частоти (до 10-12). Все це дуже гарно можна побачити на маленьких відстанях, коли співвідношення сигнал/шум має велике значення. З віддаленням дрону на значні відстані 20-80м кількість гармонік значно зменшується. Це пов'язано з особливістю розповсюдження акустичних хвиль у повітрі. Низькочастотні спектральні складові мають

менше затухання ніж високочастотні. Тому, чим далі від нас буде знаходитись дрон тим менше гармонік основної частоти обертання двигунів дрону ми будемо бачити у спектрі його сигналу.

На рисунку 2.2 наведено приклади спектральної щільності потужності для інших джерел звуку.

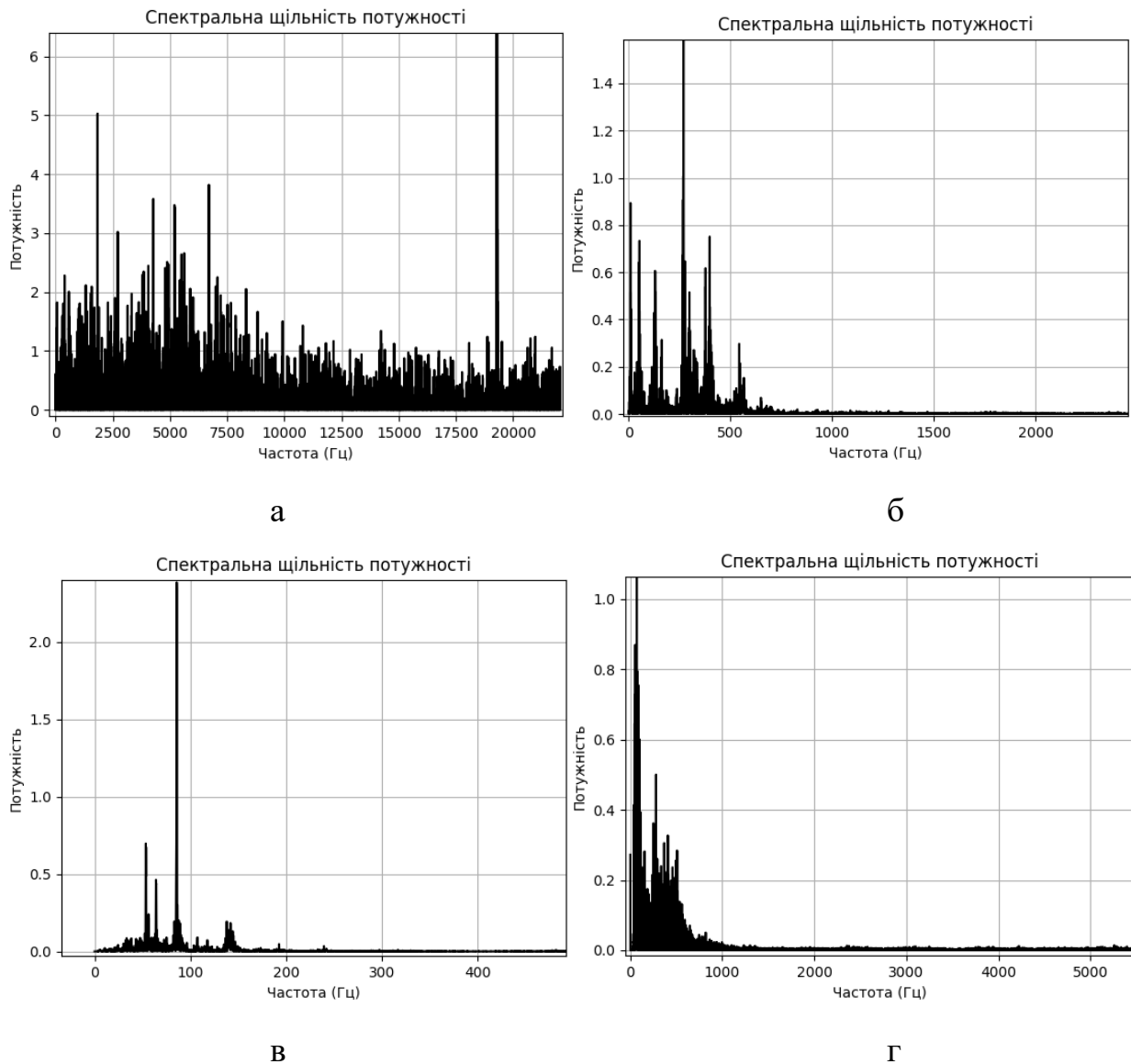
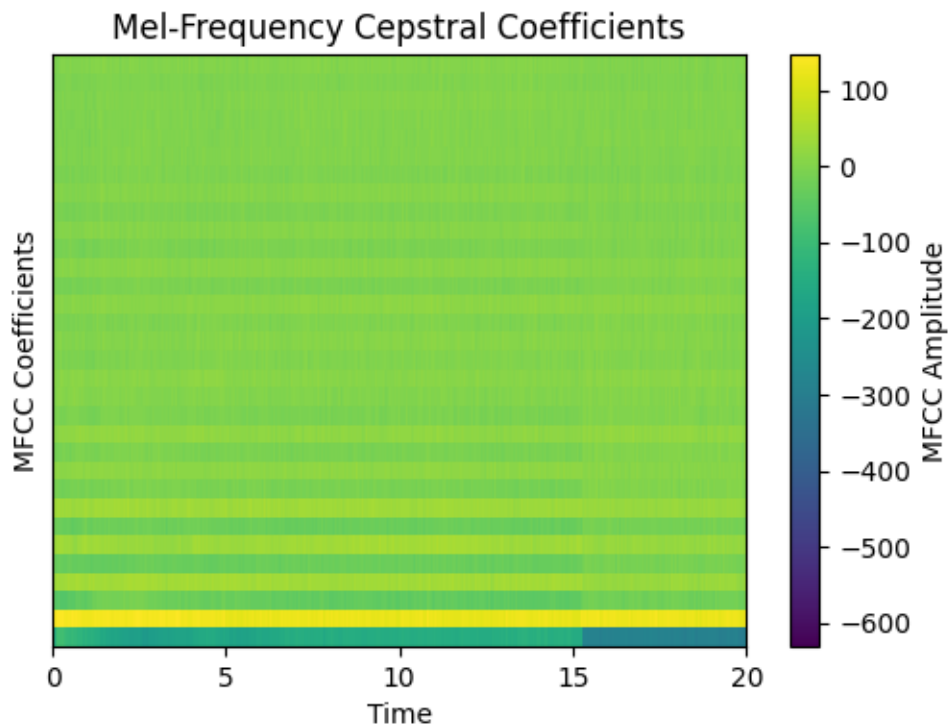


Рисунок 2.2 – Приклади спектральної щільності потужності для інших джерел звуку (а – звуки природи за містом, б – звуки голосів людей, в – звуки автотранспорту, г – звуки залізничних потягів).

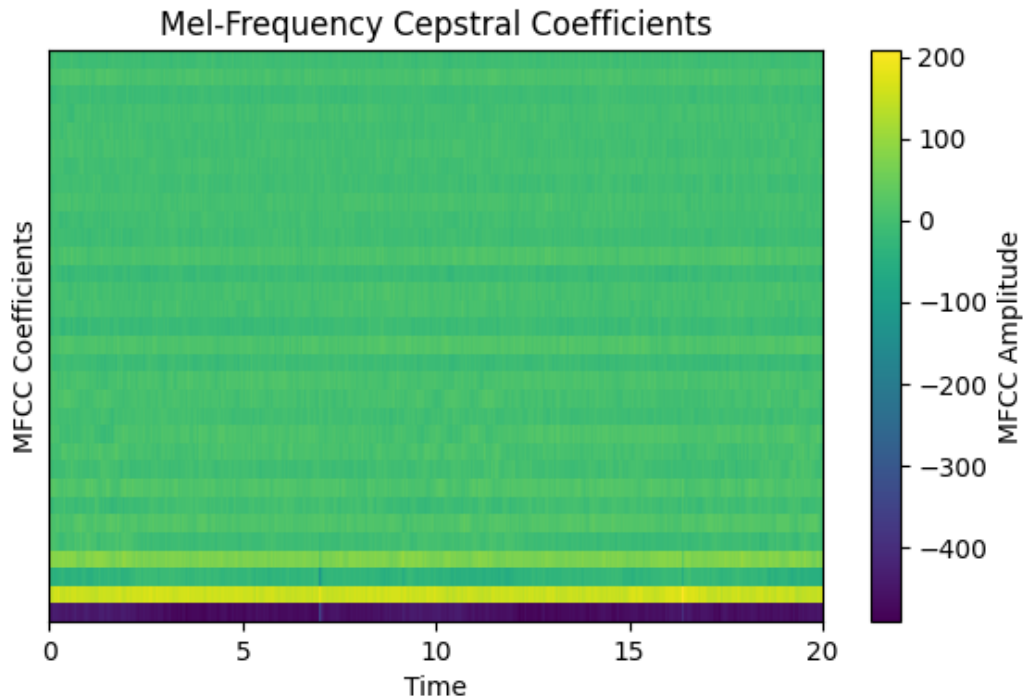
Як можна побачити з наведених на рисунку 2.2 прикладів спектральної щільності потужності потужності, вони мають значні візуальні відмінності від прикладів СЦМ на рисунку 2.1

Проведений аналіз СЦП власного випромінювання дронів та інших джерел звуку дозволяє зробити висновок про можливість якісного розпізнавання власного випромінювання дронів.

Однак СПМ власного випромінювання дронів несе занадто багато надлишкової інформації для подальшого розпізнавання . Тому було вирішено розраховувати значення MEL кепстральних коефіцієнтів за методикою, що була проаналізована у розділі 1.3. Кількість коефіцієнтів була обрана від 10 до 32. Менша кількість коефіцієнтів дає можливість підвищити швидкодію обчислень, але зменшує ефективність розрізнення векторів коефіцієнтів. Приклади зміни значень MEL кепстральних коефіцієнтів з часом у аудіо записах наведені на рисунку 2.3



а



б

Рисунок 2.3 – Приклади зміни значень MEL кепстральних коефіцієнтів з часом (а – політ дрону, б – звуки природи)

Аналізуючи приклади, що наведені на рисунку 2.3, можна побачити відмінності як у абсолютних значеннях коефіцієнтів, так і у періодичній структурі коефіцієнтів, що більш чітко виражена для дронів.

2.2 Підготовка датасету акустичних випромінювань дронів для проведення навчання нейронних мереж

Ефективність навчання будь-якої нейронної мережі залежить від кількості даних у датасеті, що використовується для навчання. Чим більше прикладів для навчання, тим краще буде результат навчання. В основному

для якісного навчання необхідний датасет із не менш ніж 300-400 прикладів на кожний клас навчання. В даному випадку ефективність навчання залежить не тільки від кількості зроблених аудіо записів польотів наших дронів, але і від різноманітності фонових звуків, типів дронів. Саме тому крім зроблених самостійно аудіозаписів були використані існуючі датасети з різноманітних ресурсів мережі Internet, як, наприклад:

- <https://github.com/saraalemedi/DroneAudioDataset>
- <https://github.com/DroneDetectionThesis/Drone-detection-dataset>
- <https://www.scidb.cn/en/detail?dataSetId=b2b4bdad91284ecaa6ffc26f919199ff>.

Всі приклади підбирались у вигляді файлів формату wav та розміщались у двох окремих папках. В першій були аудіо з записами власного акустичного випромінювання дронів в різних умовах, на різних відстанях. В другій папці були зібрані приклади різноманітних фонових звуків, що не треба розпізнавати між собою, а, лише, необхідно відносити до загального класу фонових звуків. Також було створено ще одну папку із записами польотів гелікоптерів, бо їх акустичне випромінювання хоча і відрізняється від дронів, але не дуже сильно. Ці три папки із акустичними файлами використовувались для навчання нейронних мереж. Для подальшого тестування ефективності навчання була створена ще одна папка з аудіо записами, яким відповідали вимірювання відстані між дроном та мікрофоном, щоб можна було проаналізувати ефективність навчених нейронних мереж на різних відстанях. Фактично в цій папці були пари файлів: запис аудіо та файл з результатами розрахунку дальності до дрона з часом.

3 ПРОЕКТУВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ РОЗПІЗНАВАННЯ АКУСТИЧНИХ ВИПРОМІНЮВАНЬ ДРОНІВ ТА ЇХ НАВЧАННЯ

3.1 Проектування нейронних мереж

На цей час найбільш поширеними та популярними фреймворками для проектування та навчання нейронних мереж є: TensorFlow [14] та PyTorch. Кожен з них має свої переваги та недоліки. Перевагами TensorFlow є:

- підтримка розподіленого навчання на кластерах та хмарних платформах (Google Cloud, AWS);
- оптимізований для роботи як на CPU, так і на GPU та дозволяє запускати моделі на вбудованих системах;
- гнучкість та інтеграція з мобільними пристроями;
- наявність спеціалізованих утиліт для конвертації моделей для використання на простих апаратних платформах (мікроконтролерах).

Однак є і недоліки:

- більш складний для новачків через низькорівневий синтаксис;
- більш складне налагодження через статичний граф обчислень (у версіях до TensorFlow 2.0).

До переваг PyTorch відносяться:

- простота та читабельність коду;
- більш інтуїтивний та "пітонічний" синтаксис, що дозволяє швидко створювати прототипи;
- динамічний граф обчислень (Define-by-Run);
- легка інтеграція з CUDA для прискорення обчислень;
- інтеграція з науковими бібліотеками Python (NumPy, SciPy, Pandas).

До недоліків PyTorch відносяться:

- менш оптимізований для великих виробничих систем;
- відсутність вбудованого інструменту для розгортання моделей (як TensorFlow Serving).

В проєктованій системі функцію розпізнавання акустичних випромінювань дронів буде виконувати мікроконтролер. Тому будемо використовувати фреймворк TensorFlow, щоб мати можливість сконвертувати модель нейронної мережі у бібліотечний файл, зрозумілий для мікроконтролера, а також для отримання найбільшої швидкодії обробки сигналів.

Для проєктування була обрана згортова нейронна мережа. Традиційна згортова мережа призначена для обробки зображень. Але дуже часто її використовують для розпізнавання акустичних сигналів по їх спектрограмам або по спектрограмам MEL коефіцієнтів [5,6]. В таких випадках частина спектрограми сприймається, як прямокутне зображення у відтінках сірого.

Для досліджень будемо використовувати спектрограми MEL кепстральних коефіцієнтів, побудовані для кількостей коефіцієнтів: 10,20,32. Побудова спектрограм буде проводитись на часовому інтервалі 0,5с.

$$MFCC_Matrix_Size = (N_{frames} \cdot N_{coeffs}), \quad (2.1)$$

де: N_{frames} — кількість вікон (фреймів), на які розбивається аудіосигнал.

N_{coeff} — кількість коефіцієнтів MFCC для кожного фрейму (зазвичай 13 або 20).

Кількість фреймів розраховується за формулою

$$N_{frames} = \frac{L-W}{s} + 1, \quad (2.2)$$

L — довжина аудіосигналу, семплів;

W — розмір вікна, мс;

S — крок між вікнами, мс.

Інтервал аналізу 0,5с при частоті дискретизації 44100Гц відповідає $L=22050$. Якщо взяти розмір вікна 25мс, що відповідає кількості семплів 1102, а крок між вікнами 5,5мс, що відповідає 242 семплам, то кількість фреймів буде дорівнювати

$$N_{frames} = \frac{22050-1102}{441} + 1 = 87.$$

Отже для трьох обраних значень кількості кепстральних коефіцієнтів на вхід загорткової нейронної мережі буду подаватись матриці з розмірами 10x87, 16x87, 32x87.

В нейронній мережі, що проектується, будемо використовувати на початку 2 чи 4 згорткових шара, що дозволяють виділити особливі ознаки MEL кепстральних спектрограм. Після загорткових шарів додамо повнозв'язні шари для реалізації класифікації акустичних сигналів.

Фреймворк TensorFlow для проектування пропонує наступні типи шарів:

1) Conv2D (Згортковий шар). Це основний шар для обробки зображень та акустичних сигналів. Він виконує згортку ядра (фільтра) з вхідними даними, виділяючи ключові ознаки (features). Функція реалізації цього шару має наступні вхідні параметри:

`filters` — кількість фільтрів (кількість карт ознак);

`kernel_size` — розмір фільтра (наприклад, (3, 3));

`strides` — крок згортки;

`padding` — режим (same або valid).

2) MaxPooling2D (Шар підсемплінгу). Виконує зменшення розмірності карти ознак (downsampling), виділяючи найважливіші елементи. Це

допомагає зменшити розмірність даних та уникнути перенавчання. Функція конфігурації шару має наступні вхідні параметри:

`pool_size` — розмір вікна (наприклад, (2, 2));

`strides` — крок зсуву.

3) `BatchNormalization` (Шар нормалізації). Нормалізує активації нейронів для прискорення навчання та стабілізації градієнтів.

4) `Flatten` (Розгортання вектора ознак). Перетворює 2D карту ознак у плоский вектор для подачі на повнозв'язний шар.

5) `Dense` (Повнозв'язний шар). Останній шар, який приймає розгорнутий вектор та класифікує дані.

6) `Dropout` (Шар регуляризації). Випадково вимикає частину нейронів під час навчання для запобігання перенавчанню.

Перша найпростіша нейронна мережа буде мати у вхідному загортковому шарі 32 ядра згортки і функцію активації «ReLU». Ядра мають розміри (3x3). Функція активації «ReLU» обрана з точки зору зменшення кількості обчислень. Після згорткового шару йде шар `MaxPooling2D`, що зменшить розмір карт вдвічі. Далі встановлюємо наступний згортковий шар із 64 ядрами та шар `MaxPooling2D`. Для переходу від структур матриць ознак до вектору, що треба подавати на повнозв'язний шар необхідний шар `Flatten`. За ним буде повнозв'язний шар із 1024 нейронами. Кінцевий повнозв'язний шар містить лише 3 нейрона за числом класів звуків, що будемо розпізнавати. Для уникнення перенавчання перед повнозв'язними шарами додано шар `Dropout`. Нижче наведено програмну реалізацію моделі нейронної мережі мовою Python

```
model = Sequential([
    Conv2D(32, (3, 3), activation='relu',
input_shape=(N_MFCC, MAX_PAD_LENGTH, 1)),
    MaxPooling2D((2, 2)),
```

```

Conv2D(64, (3, 3), activation='relu'),
MaxPooling2D((2, 2)),
Flatten(),
Dense(1024, activation='relu'),
Dropout(0.3),
Dense(len(CATEGORIES), activation='softmax')
])

```

Друга нейронна мережа має більш складну архітектуру. Після першого загорткового шару (32 ядра з розмірами 3x3) ми не зменшуємо розмір карт ознак, а додаємо шар BatchNormalization для нормалізації градієнтів. Другий шар аналогічний до першого. Після нього теж встановлено шар BatchNormalization. Далі йдуть шари MaxPooling2D та Dropout, щоб уникнути перенавчання. В третьому та четвертому загорткових шарах використовуються 64 ядра розмірами 3x3. Далі додані шари BatchNormalization, MaxPooling2D та Dropout. Для переходу від структур матриць ознак до вектору додано шар Flatten. За ним знаходиться повнозв'язний шар із 512 нейронами. Кінцевий повнозв'язний шар містить лише 3 нейрона за числом класів звуків, що будемо розпізнавати.

```

model = Sequential()
model.add(Conv2D(32, (3, 3), padding='same',
activation='relu', input_shape=(N_MFCC, MAX_PAD_LENGTH,
1)))
model.add(BatchNormalization())
model.add(Conv2D(32, (3, 3), padding='same',
activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.3))
model.add(Conv2D(64, (3, 3), padding='same',

```

```

activation='relu'))
model.add(BatchNormalization())
model.add(Conv2D(64, (3, 3), padding='same',
activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.5))
model.add(Flatten())
model.add(Dense(512, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(len(CATEGORIES), activation='softmax'))

```

При навчанні нейронних мереж будуть використовуватись оптимізатори Adam (Adaptive Moment Estimation) та RMSprop (Root Mean Square Propagation) [13].

Adam – поєднує переваги двох інших алгоритмів: Momentum та RMSprop. Перший метод використовує експоненційно зважене середнє попередніх градієнтів для прискорення збіжності. Другий – масштабує градієнти на основі середньоквадратичного значення їх попередніх значень, що допомагає уникнути проблем із високими або низькими темпами навчання. До переваг цього оптимізатора відносяться: автоматично адаптує темп навчання для кожного параметра, поєднує переваги SGD із моментом та RMSprop, добре працює з шумними градієнтами.

Основні рівняння Adam:

1. обчислення першого моменту m_t (експоненційно зважене середнє градієнтів)

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \quad (2.3)$$

де: β_1 - коефіцієнт експоненційного згладжування для першого моменту, "моментум", що контролює, наскільки сильно враховуються попередні градієнти в обчисленні середнього градієнта;

β_2 - коефіцієнт експоненційного згладжування для другого моменту, RMSprop-складова, що визначає, наскільки сильно враховуються попередні квадрати градієнтів;

2. обчислення другого моменту (експоненційно зважене середнє квадрату градієнтів)

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2; \quad (2.4)$$

3. коригування зміщення (оскільки початкові моменти мають зміщення до нуля)

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad \hat{v}_t = \frac{v_t}{1 - \beta_2^t}; \quad (2.5)$$

4. оновлення параметрів

$$\theta_t = \theta_{t-1} - \frac{\alpha}{\sqrt{\hat{v}_t + \epsilon}} \hat{m}_t, \quad (2.6)$$

де: α – темп навчання;

ϵ - малий доданок для уникнення ділення на нуль.

RMSprop оновлює параметри, використовуючи середньоквадратичне значення градієнтів, що згладжує оновлення та запобігає надто великим або малим змінам ваг. Оновлення середнього квадрата градієнтів реалізується за формулою

$$v_t = \beta v_{t-1} + (1 - \beta) g_t^2; \quad (2.7)$$

де: β – коефіцієнт згладжування (зазвичай 0.9).

Оновлення параметрів виконується за формулою

$$\theta_t = \theta_{t-1} - \frac{\alpha}{\sqrt{v_t + \epsilon}} g_t, \quad (2.8)$$

Навчання нейронних мереж буде проходити впродовж 100 епох. Розмір батча буде дорівнювати 8. Навчання буде проходити з контролем зменшення втрат валідації. При знаходженні найменшого значення втрат сукупність значень ваг нейронної мережі буде зберігатись автоматично. При відсутності зменшення втрат впродовж 10 епох навчання буде припинено.

3.2 Результати навчання нейронних мереж

В результаті компіляції конфігурації першої нейронної мережі було визначено кількість параметрів – 7494019. Результати навчання для першої нейронної мережі наведено на рисунку 3.1



a

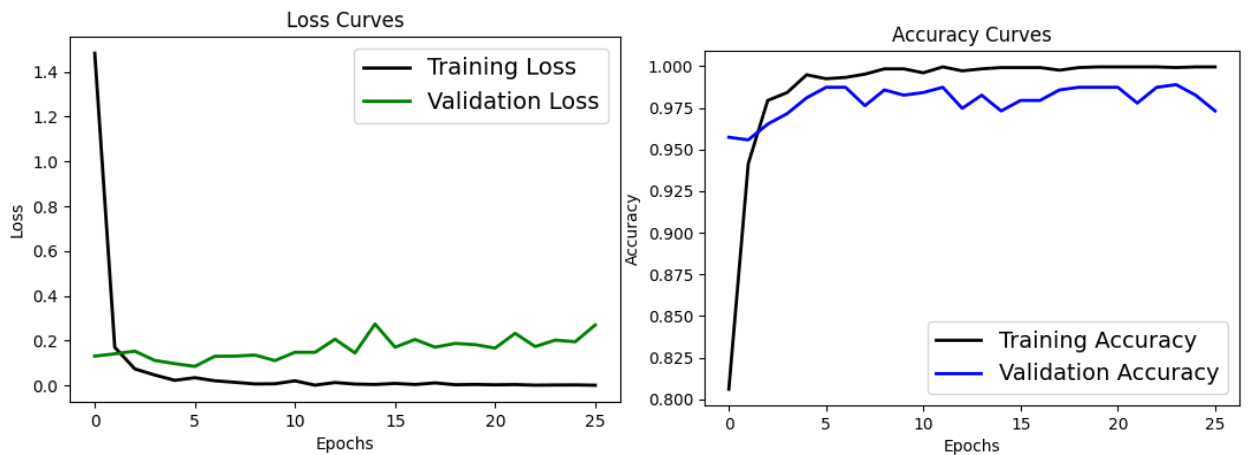
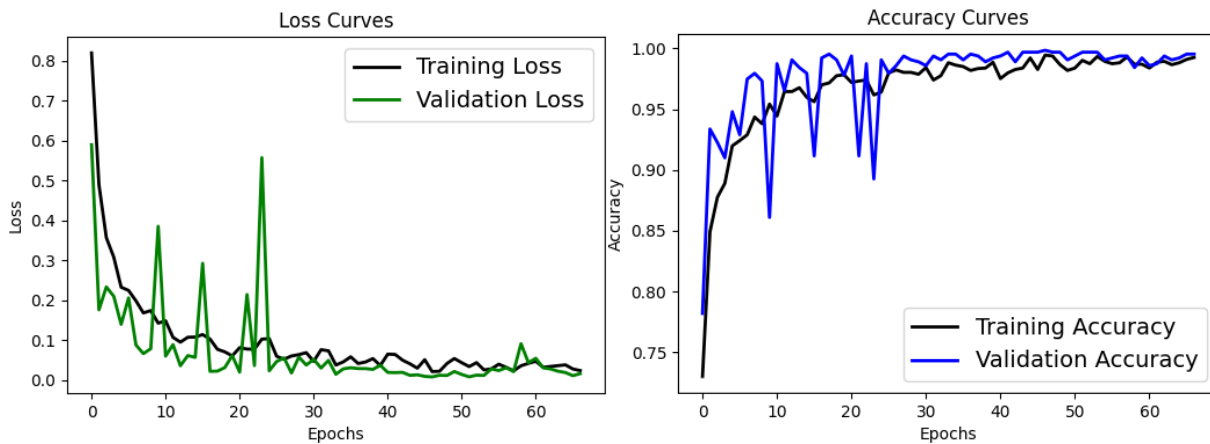
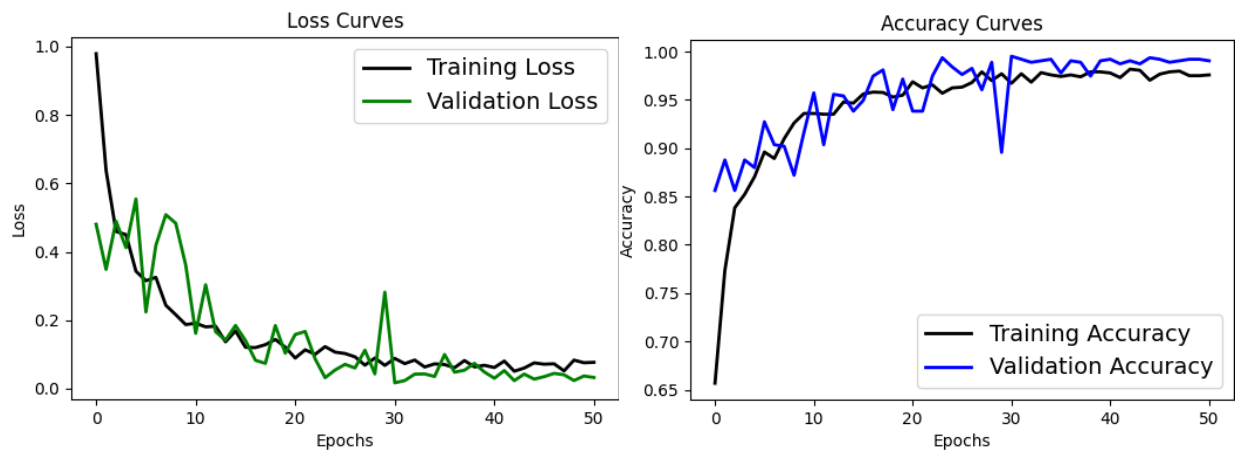


Рисунок 3.1 – Залежності втрат та точності розпізнавання від кількості епох при навчанні першої нейронної мережі (а – оптимізатор Adam, б – оптимізатор RMS)

Як можна побачити із залежностей рисунку 3.1 досягти менших значень втрат та більших значень точності можна при використанні оптимізатора RMS. Аналіз залежності втрат для валідаційної частини датасету дозволяє зробити висновок про перенавчання, бо з збільшенням кількості епох ми бачимо зріст втрат.

В результаті компіляції конфігурації другої нейронної мережі було визначено кількість параметрів – 5312739. Результати навчання для першої нейронної мережі наведено на рисунку 3.2





б

Рисунок 3.2 – Залежності втрат та точності розпізнавання від кількості епох при навчанні другої нейронної мережі (а – оптимізатор Adam, б – оптимізатор RMS)

Аналіз залежностей рисунку 3.2 показує, що використання оптимізатора Adam забезпечує менше значення втрат (0,008) порівняно з оптимізатором RMS (0.013) і, відповідно, покращити якість розпізнавання трьох обраних класів звуків.

Друга нейронна мережа забезпечує значене зменшення втрат, підвищення точності розпізнавання порівняно з першою нейронною мережею. Це пояснюється підвищенням кількості загорткових шарів та додаванням шарів, що забезпечують захист від перенавчання. Враховуючи переваги другої архітектури нейронної мережі та необхідність реалізації цієї мережі на простих апаратних платформах було вирішено скоротити кількість нейронів у Dense з 512 до 128. Після компіляції цієї архітектури було визначено, що кількість параметрів зменшилась до 1357236. Результати ефективності останньої архітектури показані на рисунку 3.3

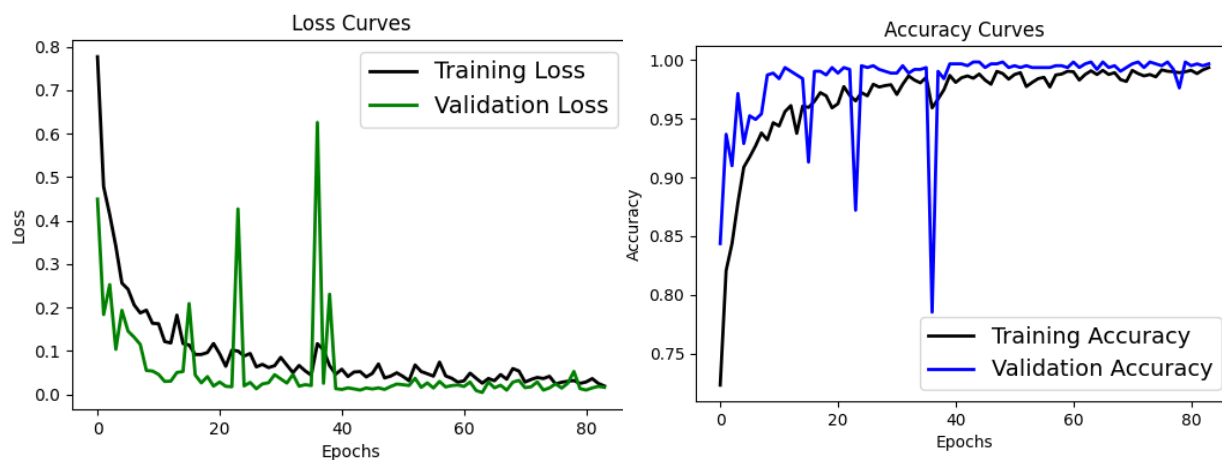


Рисунок 3.3 – Рисунок 3.2 – Залежності втрат та точності розпізнавання від кількості епох при навчанні третьої нейронної мережі (оптимізатор Adam)

Ці результати демонструють збереження ефективності розпізнавання, що зменшилась не більше ніж на 0,2%.

3.3 Результати тестування нейронних мереж

Тестування навчених нейронних мереж здійснювалось на додатковому датасеті акустичних випромінювань польотів дронів. Результати тестування першої архітектури нейронної мережі наведено на рисунку 3.4 у вигляді матриць помилок. На рисунках 3.5 та 3.6 наведені аналогічні матриці для другої та третьої архітектур. Аналіз даних, наведених на рисунках 3.4-3.6 показує, що порівняно з першою архітектурою друга та третя показують значне зменшення імовірностей помилкового розпізнавання та не розпізнавання дронів.

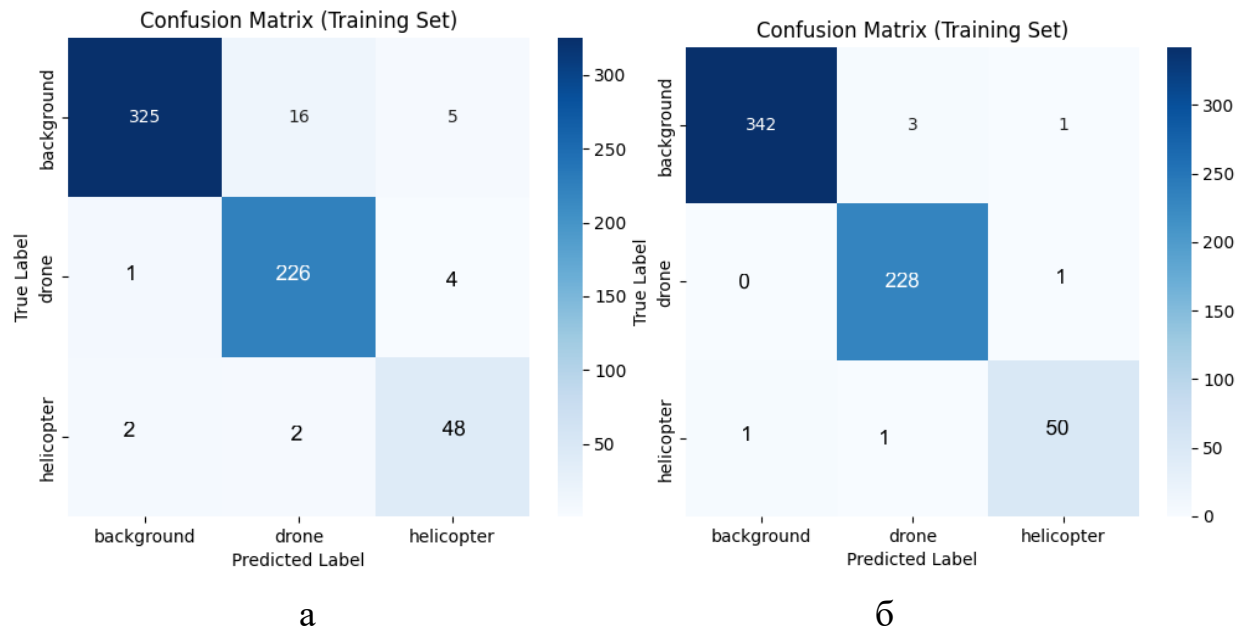


Рисунок 3.4 – Матриці помилок для першої архітектури нейронної мережі (а – оптимізатор Adam, б – оптимізатор RMS)

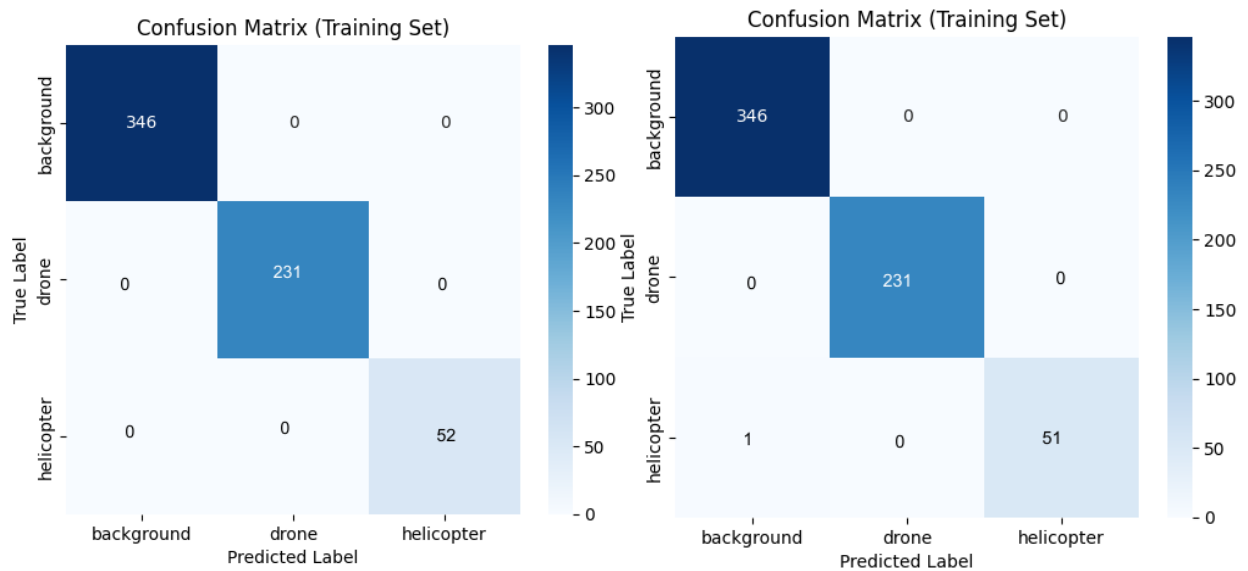


Рисунок 3.5 – Матриці помилок для другої архітектури нейронної мережі (а – оптимізатор Adam, б – оптимізатор RMS)

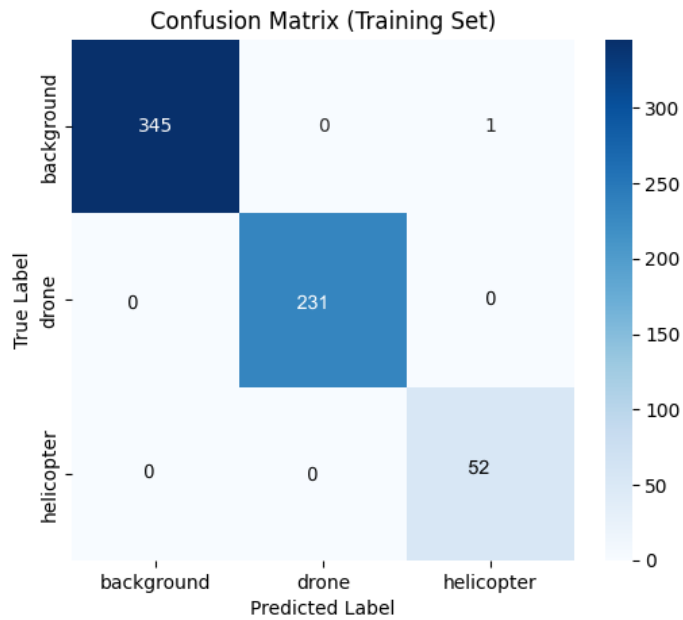


Рисунок 3.5 – Матриці помилок для третьої архітектури нейронної мережі (оптимізатор Adam)

Також було проведено аналіз якості розпізнавання дронів на різних дистанціях. Так на рисунках 3.6-3.8 показано приклад розпізнавання польоту дрону на відстанях від 5м до 100м.

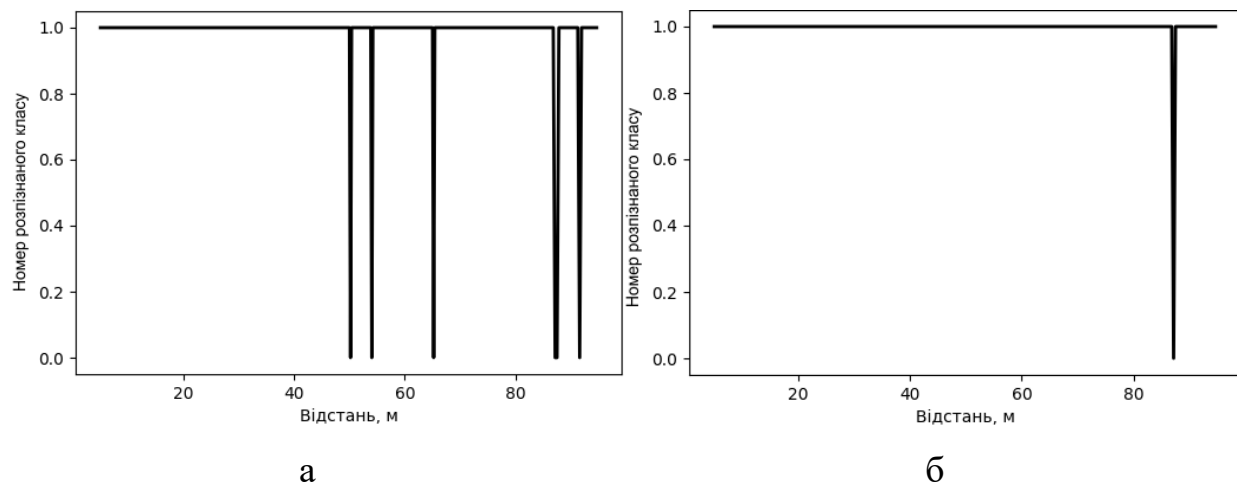


Рисунок 3.6 – Приклад розпізнавання дрону за допомогою першої нейронної мережі на різних дистанціях (а – оптимізатор Adam, б – оптимізатор RMS)

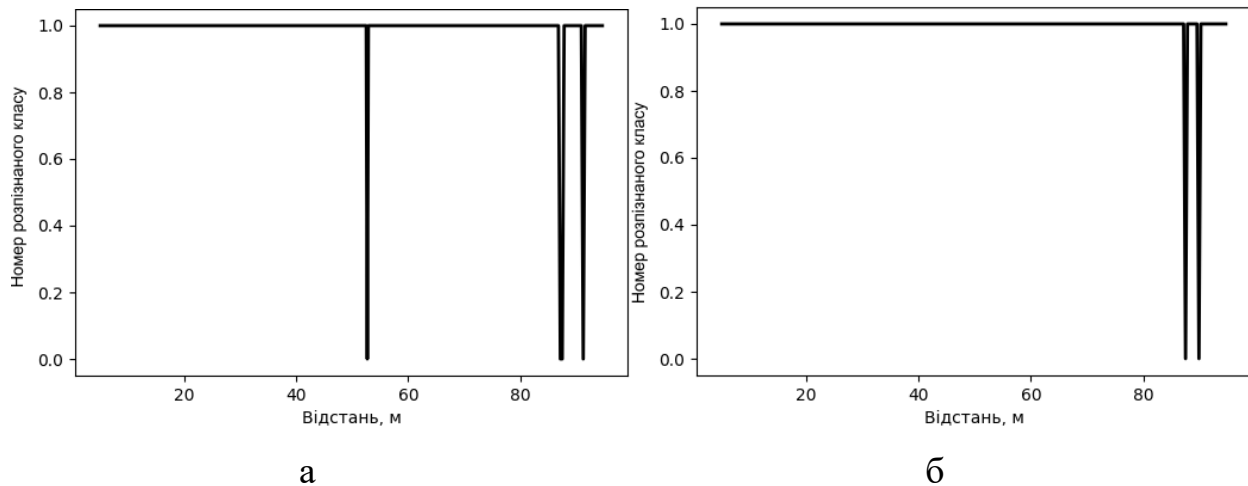


Рисунок 3.7 – Приклад розпізнавання дрону за допомогою другої нейронної мережі на різних дистанціях (а – оптимізатор Adam, б – оптимізатор RMS)

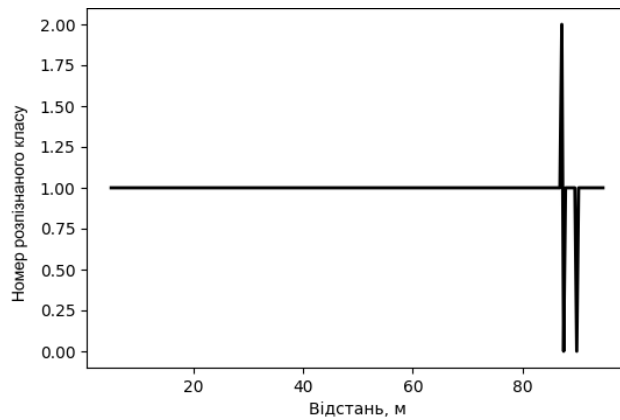


Рисунок 3.8 – Приклад розпізнавання дрону за допомогою третьої нейронної мережі на різних дистанціях (оптимізатор Adam)

Аналіз рисунків 3.6-3.8 показує, що використання оптимізатора RMS дозволяє уникнути помилок на відстанях до 80м. Третя архітектура нейронної мережі продемонструвала найбільшу кількість помилок, але це збільшення помилок не можна вважати критичним при 5 кратному зменшенні розміру моделі.

Швидкодія моделей оцінявалась на GPU Nvidia RTX305Ti. Час обробки для першої та другої нейронної мережі складав від 24 до 38мс, а для третьої від 24 до 33мс.

Таким чином, було доказано, що для практичного використання і вбудування в апаратну платформу необхідно використовувати третій варіант нейронної мережі.

4 РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ СТРУКТУРНОЇ

Запропонована електрична структурна схема пристрою розпізнавання дронів за їх акустичним випромінюванням наведена на рисунку 4.1

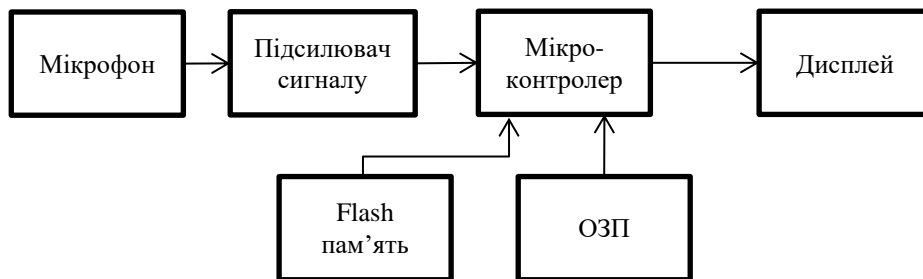


Рисунок 4.1 – Електрична структурна схема пристрою розпізнавання дронів за їх акустичним випромінюванням

Одним з найголовніших елементів цієї системи є мікрофон. Існує декілька типів мікрофонів: динамічні, конденсаторні, електретні, лінійні. Мікрофон повинен перетворювати у електричний сигнал звуки акустичного шуму гвинтів дронів. При цьому такі хвилі можуть бути, як слабкі, коли дрон знаходиться на значних відстанях, так і потужні при значному наближенні дрона. Це означає, що такий мікрофон повинен мати високу чутливість, низький рівень власних шумів та значний динамічний діапазон. Усім цим вимогам задовольняють конденсаторні мікрофони. Конденсаторні мікрофони мають: висока чутливість, широкий частотний діапазон, потребують фантомного живлення (+48V), вразливі до механічних впливів. Їх використовують у: студійному запису, конференціях, підкастах. Чутливість таких мікрофонів становить 10–50 мВ/Па, динамічний діапазон в межах 85-135дБ, рівень власних шумів 10-20дБ.

Акустичний підсилювач повинен забезпечити рівень підсилення сигналу до 3В, тобто амплітуда сигналу повинна бути $U_A=1,5В$ при мові уніполярного сигналу на вході аналого-цифрового перетворювача. Це означає, що для конденсаторного мікрофона необхідно підсилювати сигнал від $U_M=0,2-1мВ$ до 1,5В. А це означає, що коефіцієнт підсилення буде

$$K_{\Pi} = \frac{U_A}{U_M},$$
$$K_{\Pi} = \frac{1,5}{0,2 \cdot 10^{-3}} = 7500.$$

Мікроконтролер повинен мати вбудований аналого-цифровий перетворювач для перетворення аналогового сигналу в цифрову форму. Виходячи з того, що при навчанні нейронної мережі використовувалися записи акустичних сигналів з розрядністю 16 біт, то для збереження якості розпізнавання необхідна розрядність АЦП теж 16 біт. Частота дискретизації аналогового сигналу у аналого-цифрового перетворювача повина бути 44,1кГц, що відповідає записан, на базі яких проводилось навчання.

Розроблена нейронна мережа має 1357236 параметрів, кожний з яких – це число з комо, на зберігання якого потрібно 4 байта. Тому власної flash пам'яті сучасних контролерів для зберігання прошивки контролера та параметрів нейронної мережі недостатньо. Тому будемо використовувати зовнішню пам'ять на 64Мбіт. Сучасні мікросхеми пам'яті використовують інтерфейс SPI, тому мікроконтролер повинен мати вбудований інтерфейс SPI. Для реалізації запису акустичного сигналу впродовж часу 0,5с необхідно мати об'єм оперативної пам'яті 44100байт. На обчислення швидкого перетворення Фур'є, Mel кепстральних коефіцієнтів потрібно ще близько 300кбайт оперативної пам'яті. Далі зберігання кепстограми потребує 2,8кбайт, її обробка за допомогою загорткових шарів нейронної мережі потребує 250кбайт. У повнозв'язних шарах нейронної мережі для обчислень

необхідно теж не менш 100кбайт. До цього слід додати необхідність отримати максимальну швидкодію і завантажити ваги нейронної мережі у оперативну пам'ять. Тоді потрібний загальний об'єм оперативної пам'яті 6,2Мбайт. Існує багато сучасних мікроконтролерів, на яких реалізуються алгоритми розпізнавання, але робота у реальному часі можлива лише на спеціалізованих мікронтролерах, архітектура яких оптимізована для роботи нейронних мереж та виконання функцій цифрової обробки сигналів із спеціалізованих бібліотек цифрової обробки. Саме тому необхідно обрати мікроконтролер, що задовольняє усім переліченим вимогам.

Дисплей повинен відображати інформацію про результати розпізнавання: наявність або відсутність дрону, точність розпізнавання. Розмір такого дисплея не є критичним для цього пристроя. Тому будемо використовувати кольоровий дисплей з розміром 3,2 дюйма.

5 РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ ПРИНЦИПОВОЇ ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ МІКРОКОНТРОЛЕРА

5.1 Вибір мікрофону та мікрофонного підсилювача для акустичних вимірювань

Для збереження ефективності акустичних вимірювань обираємо мікрофон Superlux ECM-999, що використовувався для запису акустичних сигналів перед навчанням нейронних мереж. Його характеристики наведені у таблиці 5.1

Таблиця 5.1 – Характеристики мікрофону Superlux ECM-999

Тип мікрофона:	Конденсаторний вимірювальний
Діаграма спрямованості:	Ненаправлений
Діапазон частот :	20 Гц – 20 кГц
Чутливість:	-37dBV/Pa (14mV/Pa)
Опір R :	200 Ом
Рекомендований опір навантаження:	3 кОм
Максимальний рівень звукового тиску :	132 дБ
Фантомне живлення :	9-52 В

Для обраного мікрофона чутливість складає -37дБВ/Па, тобто при 94 дБ SPL, а сигнали на рівні шопоту мають 30 дБ SPL. Розрахуємо звуковий тиск за формулою (5.1).

$$P = P_0 \cdot 10^{\frac{SPL}{20}},$$

де P_0 – опорний тиск ($20 \cdot 10^{-6}$), Па.

$$P = 20 \cdot 10^{-6} \cdot 10^{\frac{30}{20}} = 0.632 \text{ мПа.}$$

Тоді чутливість мікрофона буде

$$U = 14 \cdot 10^{-3} \cdot 0.632 \cdot 10^{-3} = 8.8 \text{ мкВ.}$$

Для підсилення сигналу від мікрофону до рівня на вході аналого-цифрового перетворювача мікроконтролера будемо використовувати спеціалізований підсилювач на мікросхемі PGA2500. Рівень сигналу на вході АЦП мікроконтролера зазвичай досягає 3В. Тоді максимальний коефіцієнт підсилення повинен бути

$$K_{\text{п}} = \frac{3}{8.8} \cdot 10^{-6} = 390909 = 110 \text{ дБ}$$

Обрана мікросхема повністю задовольняє вимогам, бо призначена для роботи з конденсаторними мікрофонами та має коефіцієнт підсилення до 120дБ з регулюванням з кроком по 3дБ. Принципова схема підключення мікросхеми PGA2500 наведена на рисунку 5.1

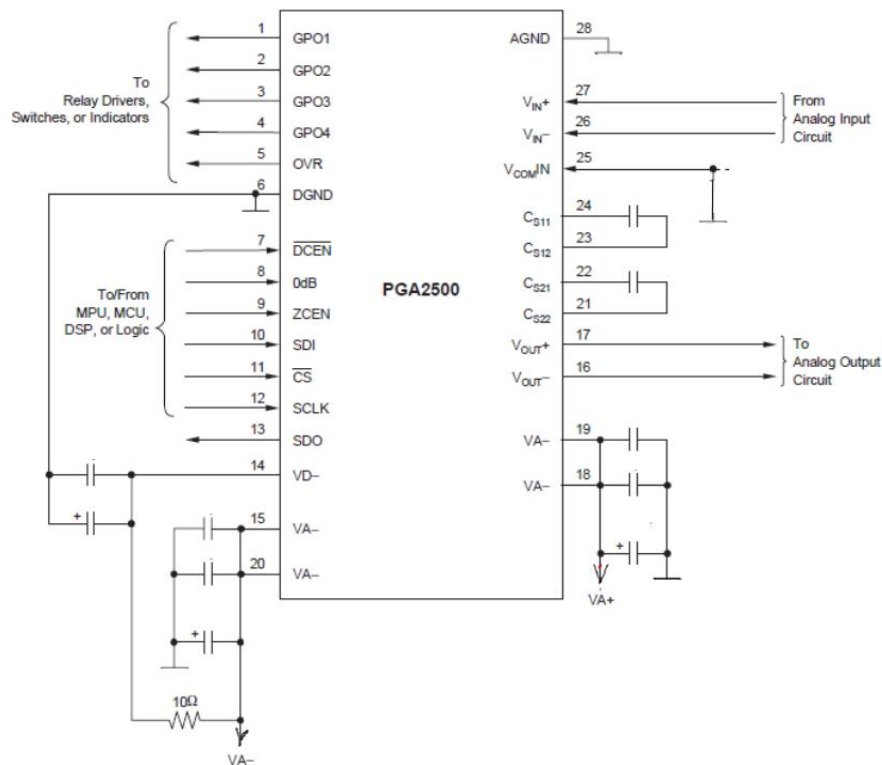


Рисунок 5.1 – Схема електрична принципова підключення PGA2500

На входи V_{in+} , V_{in-} подається сигнал від мікрофона, з виходів V_{out+} , V_{out-} підсилений сигнал буде подаватись на аналого-цифровий перетворювач мікроконтролера. Лінії SDO, SDI, CS, SCLK, DCEN є лініями інтерфейса SPI, що використовується для керування підсиленням акустичного сигналу. Завдяки цим лініям буде реалізовано систему автоматичного керування підсиленням, щоб отримувати постійно амплітуду акустичного сигналу на вході АЦП близькою до максимального значення напруги на вході АЦП.

5.2 Вибір керуючого мікроконтролеру

У якості мікроконтролеру, що буде керувати вимірюваннями та розпізнавати акустичне випромінювання дрону обираємо спеціалізований мікроконтролер від фірми STMicroelectronics – STM32MP157. STM32MP157 — це потужний гібридний мікроконтролер + процесор від STMicroelectronics, який поєднує ARM Cortex-A7 + Cortex-M4. Він ідеально підходить для складних застосувань, зокрема й розпізнавання зображень. Процесорна архітектура: 2 ядра ARM Cortex-A7 до 650 МГц працюють під Linux (Debian, Yocto тощо), 1 ядро ARM Cortex-M4 до 209 МГц — для real-time задач (FreeRTOS, bare metal). Cortex-A7 здатен запускати TensorFlow Lite, OpenCV, ONNX Runtime. Процесор підтримує NEON SIMD інструкцій для прискорення математичних операцій. Можна виконувати inference на CNN (convolutional neural network) у реальному часі, при зниженій роздільності або оптимізованій моделі (quantized). В нашому випадку розмір матриць MEL коефіцієнтів невеликий, що дозволяє спокійно реалізувати обробку. Принципова схема підключення до мікроконтролера периферійних мікросхем наведена на рисунку 5.2

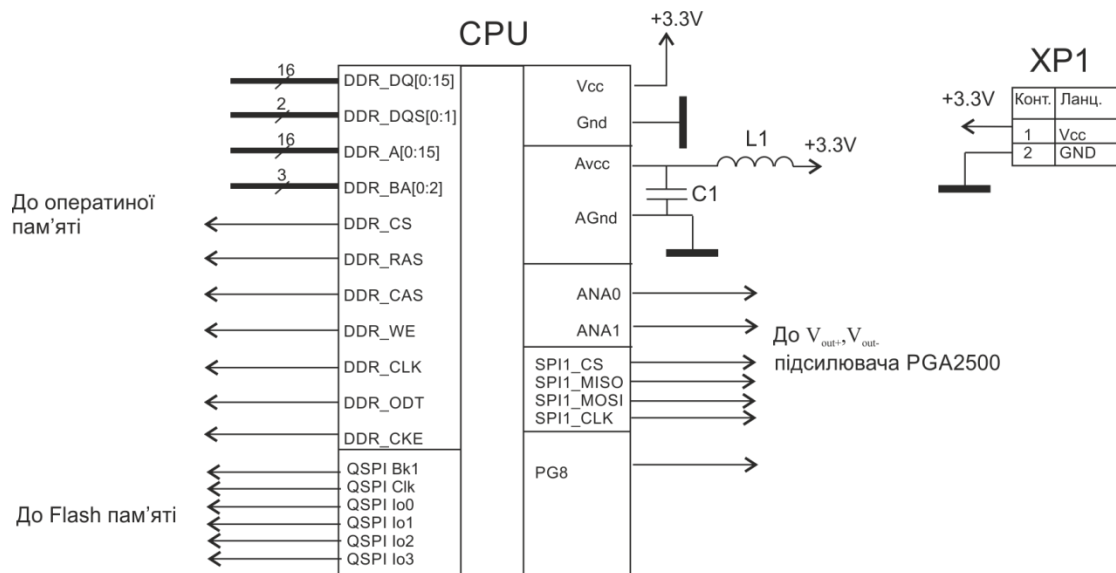


Рисунок 5.2 – Схема електрична принципова підключення зовнішніх елементів до мікроконтролера

Живлення мікроконтролера здійснюється подачею напруги 3.3В через роз'єм XP1. Додатково живлення подається на виводи Avcc, Agnd, що відповідають живленню вбудованого аналого-цифрового перетворювача. Для більш якісного згладжування живлення вбудованого АЦП використовується фільтр на елементах L1C1. Аналого-цифровий перетворювач працює в диференціальному режимі, тобто дозволяє подавати на входи ANA0, ANA1 диференціальний сигнал після підсилювача PGA2500. Керування підсилювачем здійснюється через лінії інтерфейса SPI1 (SPI1_CS, SPI1_CLK, SPI1_MOSI, SPI1_MISO).

Після розпізнавання отримана інформація про результат розпізнавання буде виводитись на графічний дисплей з SPI інтерфейсом. Тому для підключення дисплея буде використовуватись знов інтерфейса SPI1 (SPI1_CLK, SPI1_MOSI, SPI1_MISO) разом з іншим сигналом вибору мікросхеми дисплея через лінію PG8.

Для роботи нейронної мережі, а також для обчислення MEL коефіцієнтів необхідний великий об'єм оперативної пам'яті типу DDR.

Підключення цієї пам'яті буде здійснюватись через спеціалізований інтерфейс DDR: DDR_DQ[0:15] — шина даних (двунаправлена), DDR_DQS[0:1]— стробуючі динії, DDR_A[0:14] — адресна шина, DDR_BA[0:2] — шина вибору банків пам'яті, DDR_CS, DDR_RAS, DDR_CAS, DDR_WE — керуючі сигнали, DDR_CLK — дифференційований тактовий сигнал, DDR_ODTDDR_CKE — допоміжні сигнали. Для зберігання коефіцієнтів нейронної мережі буде використовуватись flash пам'ять, що буде підключатись через інтерфейс DQSPI (Bk1, Clk, Io0, Io1, Io2, Io3).

5.3 Вибір мікросхеми flash пам'яті

MX25L51245G — це серійна NOR Flash пам'ять від Macronix, що працює через інтерфейс SPI / Quad SPI (QSPI). Обсяг пам'яті мікросхеми 512 Мбіт (64 МБ), інтерфейс SPI / Quad SPI, частота до 133 МГц, напруга живлення 2.7 В – 3.6 В, розмір сторінки для запису 256 байт. Завдяки підтримці інтерфейсу Quad SPI за один такт передаються або приймаються одразу 4 біта. Підключення мікросхеми flash пам'яті до мікроконтролеру реалізується відповідно до таблиці 5.2

Таблиця 5.2 – Підключення мікросхеми flash пам'яті до мікроконтролеру

MX25L51245	STM32MP157 (QSPI)	Примітка
SCLK	QSPI_CLK	Тактовий сигнал
CS#	QSPI_Bk1	Chip Select (активний «0»)
IO0 (MOSI)	QSPI_IO0	Передача даних
IO1 (MISO)	QSPI_IO1	Прийом даних
IO2	QSPI_IO2	Використовується в Quad SPI
IO3	QSPI_IO3	Використовується в Quad SPI
VCC	3.3 В	Живлення
GND	GND	Земля

5.4 Вибір мікросхеми оперативної пам'яті

У якості оперативної пам'яті обираємо мікросхему MT41K256M16HA. Це мікросхема DDR3L SDRAM від Micron, часто використовується у вбудованих системах з контролером пам'яті. Її обсяг $256\text{М} \times 16 = 4$ Гбіт або 512 Мбайт, шина даних 16 біт, частота передачі даних до 800МГц, напруга живлення 1.35 В (DDR3L), організована у банків 8 банків. Назви ліній підключення до мікроконтролера наведені у таблиці 5.3

Таблиця 5.3 – Назви ліній підключення оперативної пам'яті до мікроконтролера та їх призначення

Назва сигналу	Призначення
DQ[0:15]	Шина даних
DQS[0]	Data strobe (для синхронізації)
A[0:14]	Адресна шина
BA[0:2]	Bank Address (вибір банку)
RAS#, CAS#, WE#	Команди доступу
CS#	Вибір чипа
CK, CK#	Тактовий сигнал (диференційний)
RESET#	Скидання
ODT	On-Die Termination
CKE	Clock Enable
VREF	Опорна напруга
VDD/VDDQ	Живлення 1.35 В
VSS/VSSQ	Земля

5.5 Вибір дисплею

У якості дисплея обираємо дисплей на базі контролера ILI9341. Дисплей ILI9341 — це популярний TFT-дисплей з інтерфейсом SPI, який широко використовується для проєктів з мікроконтролерами. STM32 легко підключається до дисплея через інтерфейс SPI.

Розроблений алгоритм перетворення акаустичних сигналів ронів у цифрову форму та розпізнавання таких сигналів наведено на рисунку 5.4

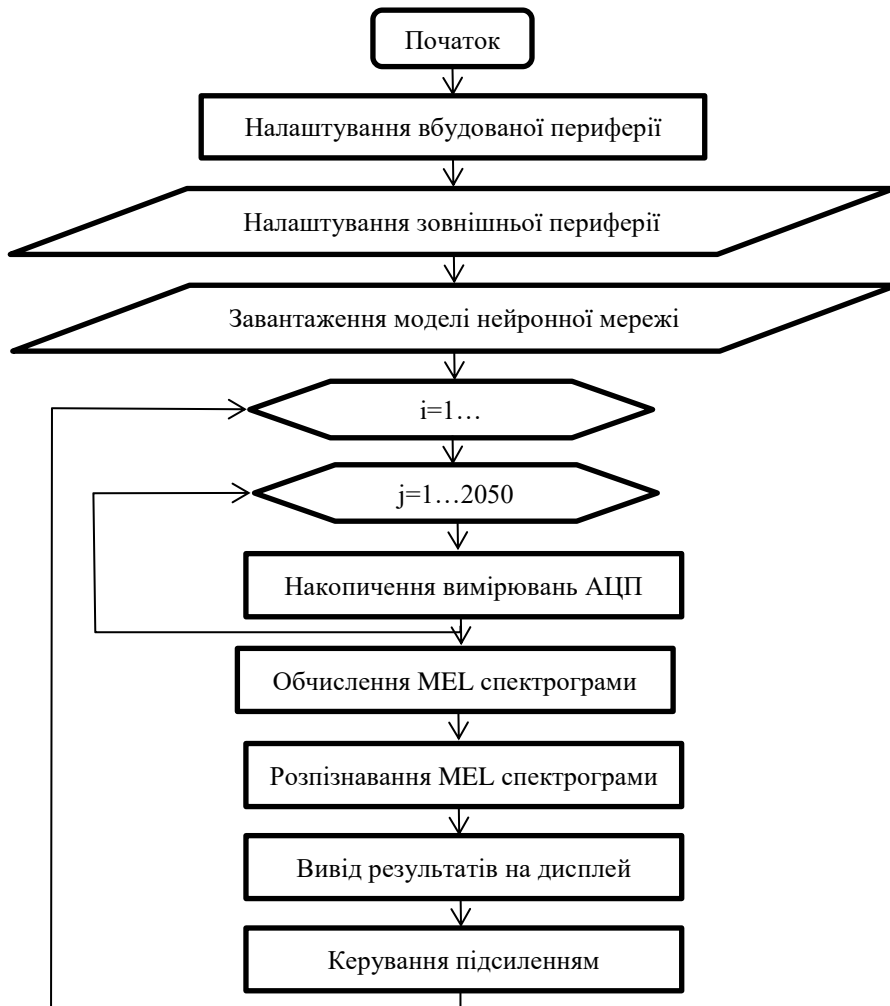


Рисунок 5.4 – Алгоритм роботи пристрою розпізнавання акустичного випромінювання дрона

При вмиканні живлення мікроконтролера починається налаштування вбудованої периферії. Налаштовується другий таймер для забезпечення періодичних аналого-цифрових перетворень з частотою 44100Гц. На таймер подається тактова частота $f_1=100\text{МГц}$, період вимірювань $T_b=1/44100=2,26 \cdot 10^{-5}$. За час періода вимірювань на таймер подається кількість тактів

$$N = T_b \cdot f_T,$$
$$N = 2,26 \cdot 10^{-5} \cdot 10^8 = 2267$$

Таймер генерує події для автоматичного запуску аналого-цифрового перетворювача. Далі налаштовується вбудований АЦП. Він автоматично запускається від таймера і разом з контролером DMA забезпечує зберігання результатів перетворення у масив в оперативній пам'яті. Розмір цього масиву (буфера) дорівнює 88200. Переривання від контролера DMA генеруються при заповненні першої та другої половини буфера. Коли заповнюється перша половина буфера, обробляється друга і навпаки. Далі налаштовується інтерфейс SPI для взаємодії з зовнішньою flash пам'яттю та підсилювачем акустичного сигналу, інтерфейс DDR для роботи із зовнішньою оперативною пам'яттю. Після цього виконується налаштування дисплею для виводу на нього результатів розпізнавання. Далі у оперативну пам'ять завантажуються ваги нейронної мережі. Після цього починається нескінчений цикл роботи пристрою. В середині циклу є вбудований цикл із 2050 ітерацій, що забезпечує накопичування 2050 вимірювань аналогового сигналу (половини буфера). Після заповнення половини буфера обчислюється MEL спектрограма, виконується її розпізнавання за допомогою нейронної мережі та вивід результатів розпізнавання на дисплей. На основі 2050 результатів вимірювань обчислюється динамічний діапазон вхідного сигналу та коректується підсилення акустичного сигналу мікросхеми PGA2500.

5.7 Розробка програмного забезпечення

Програма була розроблена з використанням середовища STM32CubeIDE. Повний текст програми наведено у додатку В. Розглянемо основні функції цієї програми для обчислення MEL спектрограми та її

розпізнавання за допомогою нейронної мережі. Спочатку створюється група констант для обчислення MEL спектрограми

```
#define FRAME_SIZE    512 – кількість відліків для обчислення коефіцієнтів MEL
#define MEL_BINS      32 – кількість MEL коефіцієнтів
#define MEL_FRAMES    87 – кількість стовпців матриці MEL коефіцієнтів
#define SAMPLE_RATE   44100 – частота дискретизації сигналу
```

В основу обчислення MEL кепстральних коефіцієнтів покладені такі операції, як швидке перетворення Фур'є, обчислення модуля спектра, MEL фільтрація, та дискретне косинусне перетворення (DCT). Всі ці операції реалізуються з використанням сучасної бібліотеки `arm_math.h`. З використанням цієї бібліотеки було створено функцію обчислення вектора MEL коефіцієнтів `void compute_mel_spectrum(const float* signal, float mel_output[MEL_BINS])`, вхідними параметрами якої є масив результатів аналого-цифрового перетворення та масив для результатів обчислення MEL коефіцієнтів. У цій функції спочатку сигнал `signal` обробляється віконною функцією `window`

```
for (int i = 0; i < FRAME_SIZE; ++i)
    frame[i] = signal[i] * window[i];
```

Далі обчислюється швидке перетворювання Фур'є за допомогою функції `arm_rfft_fast_f32` із бібліотеки `arm_math.h`. Результат обчислення – масив `fft_out`

```
arm_rfft_fast_f32(&fft_instance, frame, fft_out, 0);
```

Після цього обчислюється модуль спектру

```
arm_cmplx_mag_f32(fft_out, spectrum, FRAME_SIZE / 2);
```

Далі виконується MEL-фільтрація

```
memset(mel_banks, 0, sizeof(mel_banks));
```

```

for (int m = 0; m < MEL_BINS; ++m)
{
    for (int k = 0; k < FRAME_SIZE / 2; ++k)
    {
        float w = mel_filter_weight(m, k, FRAME_SIZE / 2);
        mel_banks[m] += w * spectrum[k];
    }
    mel_banks[m] = logf(mel_banks[m] + 1e-6f); // log scale
}

```

В кінці обчислюється DCT

```

for (int i = 0; i < MEL_BINS; ++i)
{
    mel_output[i] = 0.0f;
    for (int j = 0; j < MEL_BINS; ++j)
    {
        mel_output[i] += mel_banks[j] * cosf(M_PI * i * (2 * j + 1) / (2.0f *
MEL_BINS));
    }
}

```

В головній програмі в циклі виконується виклик функції `compute_mel_spectrum`. В кожній ітерації циклу ми отримуємо вектор MEL коефіцієнтів і формуємо з таких векторів матрицю розміром 32x87, що далі подається на нейронну мережу.

```

for (int i = 0; i < MEL_FRAMES; ++i)
{
    compute_mel_spectrum(&signal[i * 256], mel_matrix[i]);
}

```

Для розпізнавання отриманої MEL спектрограми використовуються бібліотеки `micro_interpreter.h`, `schema_generated.h`, `all_ops_resolver.h`, `version.h` із спеціалізованого фреймворку `tensorflow lite`. Завантаження моделі виконується з використанням методу `GetModel`, вхідним параметром котрої є адреса на модель у flash пам'яті

```
const tflite::Model* model = tflite::GetModel(g_model_data);
```

Далі створюється оператор `static tflite::AllOpsResolver resolver` та виконується інтерпретування завантаженої моделі

```
tatic tflite::MicroInterpreter interpreter(model, resolver, tensor_arena, TENSOR_ARENA_SIZE);
```

Після цього створюється вхідний шар та вихідний масив результатів розпізнавання

```
TfLiteTensor* input = interpreter.input(0);
```

```
TfLiteTensor* output = interpreter.output(0);
```

Далі подаємо вхідну матрицю MEL коефіцієнтів на вхід нейронної мережі

```
for (int i = 0; i < MEL_ROWS; ++i)
```

```
{
```

```
    for (int j = 0; j < MEL_COLS; ++j)
```

```
    {
```

```
        input->data.f[i * MEL_COLS + j] = mel_input[i][j];
```

```
    }
```

```
}
```

Наступним кроком робимо розпізнавання за допомогою метода `Invoke`

```
interpreter.Invoke()
```

Для зчитування вихідного значення на i -му нейроні вихідного шару використовуємо звертання до даних `output->data.f[i]`. Номер вихода, на якому отримано максимальне значення відповідає номеру класу розпізнаного звуку.

ВИСНОВОК

В ході дипломної роботи вирішувалась сучасна науково-практична задача дистанційного розпізнавання акустичних випромінювань дронів. При польоті дрони завжди створюють акустичні випромінювання при обертанні пропелерів, що дозволяє виявляти дрони в складних умовах (низька висота, ніч і т.д.). В результаті аналізу наукових публікацій з тематики дипломної роботи був обраний метод на основі розпізнавання спектрограм MEL коефіцієнтів з використанням згорткової нейронної мережі. В роботі було створено датасет записів акустичних випромінювань дрона, гелікоптера та фонових звуків. За допомогою створеного датасета були навчені 3 сконструйованих нейронних мережі з різною кількістю загорткових та повнозв'язних нейронних шарів. Результати навчання та тестування навченої нейронної мережі показали високу ефективність розпізнавання на дуальностях до 95м. Для практичної реалізації пристроя розпізнавання акустичних випромінювань дронів було спроектовано і протестовано архітектуру, що мала лише 2 повнозв'язних шара та 6 згорткових. Ця архітектура мала найменшу кількість параметрів та практично не поступалась за ефективністю розпізнавання іншим протестованим архітектурам. Для практичної реалізації пристроя розпізнавання був обран спеціалізований мікроконтролер STM32MP157, що має спеціалізовану архітектуру для реалізації роботи нейронних мереж у реальному часі. З його використанням розроблено електричну структурну та принципову схеми пристроя перетворення акустичних випромінювань дронів у цифрову форму, обчислення MEL спектрограми та розпізнавання.

Таким чином мета та завдання дипломної роботи виконані в повному обсязі. Наукової новизною отриманих результатів є розробка архітектури

нейронної мережі, що дозволяє ефективно розпізнавати акустичні випромінювання дронів і може бути реалізована на апаратних платформах з обмежаними апаратними ресурсами.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Mădălin Dombrovschi, Marius Deaconu, Laurentiu Cristea, Tiberius Florian Frigioescu, Grigore Cican, Gabriel-Petre Badea, George Totu. Acoustic Analysis of a Hybrid Propulsion System for Drone Applications. *Acoustics* 2024, 6(3), 698-712; <https://doi.org/10.3390/acoustics6030038>
2. Vladimir M Kartashov, Volodimir M Oleynikov, SA Sheyko, Stanislav I Babkin, IV Koryttsev, OV Zubkov, MA Anokhin Information characteristics of sound radiation of small unmanned aerial vehicles *Telecommunications and Radio Engineering* Volume 77, Issue 10, 2018, pp. 915-924 DOI: 10.1615/TelecomRadEng.v77.i10.70
3. Kartashov V., Oleynikov V., Koryttsev I., Zubkov O., Babkin S., Sheiko S. Processing and Recognition of Small Unmanned Vehicles Sound Signals *International Scientific-Practical/ Conference on Problems of Infocommunications Science and Technology, PIC S and T 2018 - Proceedings* 31 January 2019, Pages 392-396
4. Esraa Othman, Iva Cibilić, Vesna Poslončec-Petrić, Dina Saadallah Investigating Noise Mapping in Cities to Associate Noise Levels with Sources of Noise Using Crowdsourcing Applications *Urban Sci.* 2024, 8(1), 13; <https://doi.org/10.3390/urbansci8010013>
5. Zrar K.A., Abdulbasit K.A. Mel Frequency Cepstral Coefficient and its Applications: A Review/ *IEEE Access*, 2022, Vol.10 PP.122136 – 122158
6. Sithara A., Abraham T., Mathew D. Study of MFCC and IHC Feature Extraction Methods With Probabilistic Acoustic Models for Speaker Biometric Applications/ *Procedia Computer Science*. Vol. 143, 2018, PP. 267-276

7. Elhaik E. Principal Component Analyses (PCA)-based findings in population genetic studies are highly biased and must be reevaluated/ Open access. Sci Rep 12, 2022. <https://doi.org/10.1038/s41598-022-14395-4>
8. Mishra S., Sarkar U., Taraphder S., Datta S. Principal Component Analysis/ International Journal of Livestock Research. 2017. Vol.7, N.5, PP. 60-78. DOI:10.5455/ijlr.20170415115235
9. Purwono P., Ma'arif A., Rahmaniar W., Imam H. Understanding of Convolutional Neural Network (CNN): A Review/ International Journal of Robotics and Control Systems. 2022. Vol.2, N.4, PP.739-748.
- 10.Indolia S., Goswami A.K., Mishra S.P., Asopa P. Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach/ Procedia Computer Science. 2018, Vol.132, PP. 679-688
- 11.Fazal M.A., Baig M.A., Manj W.A., Faraz Z., Mallah G.A. Implementation of Deep Learning for Acoustic Classification/ Journal of Xidian University. 2023, Vol. 17, Iss. 8, PP.1653-1673
- 12.Pham L., Ngo D., Salovic D.^c, Jalali A., Schindler A., Nguyen P. X., Tran K. , Vu H. Lightweight deep neural networks for acoustic scene classification and an effective visualization for presenting sound scene contexts/ Applied Acoustics. 2023, Vol.211, PP.723-731
- 13.Moinul Islam, Monjurul Haque, Saiful Islam, Zesun Ahmed Mia, S. M. A. Mohaiminur Rahman DCNN-LSTM Based Audio Classification Combining Multiple Feature Engineering and Data Augmentation Techniques Intelligent Computing & Optimization Springer, Cham 2021, volume 371
- 14.Pang B., Nijkamp E., Nian Y.W. Deep Learning With TensorFlow: A Review/ Journal of Educational and Behavioral Statistics. 2019. Vol. 45, Iss. 2., PP.415-421 <https://doi.org/10.3102/107699861987276>

- 15.Soujanya B., Sitamahalakshmi T. Optimization with ADAM and RMSprop in Convolution neural Network (CNN): A Case study for Telugu Handwritten Characters/ International Journal of Emerging Trends in Engineering Research. 2020.,Vol. 8. No. 9. PP.5116-5121
- 16.PGA2500: Digitally Controlled Microphone Preamplifier datasheet (Rev. A) [Електронний ресурс]. – Режим доступу: https://www.ti.com/lit/ds/symlink/pga2500.pdf?ts=1745958437229&ref_url=https%253A%252F%252Fwww.google.com%252F
- 17.STM32MP1 series. [Електронний ресурс]. – Режим доступу: <https://www.st.com/en/microcontrollers-microprocessors/stm32mp1-series.html>
- 18.Методичні вказівки до оформлення кваліфікаційних робіт (проектів) для здобувачів усіх форм навчання спеціальностей 171 Електроніка / Упоряд.: О.В. Зубков, О.В. Воргуль. Харків: ХНУРЕ, 2024. 26 с.