Jyothi Hills, Panjal Road,
Vettikattiri PO, Cheruthuruthy , Thrissur,
Kerala 679531

NAAC
NATIONAL ASSESSMENT AND ACCREDITATION COUNCIL

Jyothi Engineering College
NAAC Accredited College with NBA Accredited Programmes*
Approved by AICTE & affiliated to APJ Abdul Kalam Technological University
A CENTRE OF EXCELLENCE IN SCIENCE & TECHNOLOGY BY THE CATHOLIC ARCHDIOCESE OF TRICHUR
JYOTHI HILLS, VETTIKATTIRI P.O, CHERUTHURUTHY, THRISSUR. PIN-679531 PH : +91- 4884-259000, 274423 FAX : 04884-274777
NBA accredited B.Tech Programmes in Computer Science & Engineering, Electronics & Communication Engineering, Electrical & Electronics Engineering and Mechanical Engineering valid for the academic years 2016-2022. NBA accredited B.Tech Programme in Civil Engineering valid for the academic years 2019-2022.

Audio Based Drone Detection and Identification using Deep Learning

MAIN PROJECT REPORT

AKSHAYA RAJ	**(JEC16CS006)**
ALWIN JOSEPH	**(JEC16CS014)**
ARCHANA VENUGOPAL	**(JEC16CS031)**
JACOB B ALAPPAT	**(JEC16CS053)**

in partial fulfillment for the award of the degree
of

BACHELOR OF TECHNOLOGY (B.Tech)

in

COMPUTER SCIENCE & ENGINEERING

of

A P J ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Under the guidance of

Dr. VINITH R

DECEMBER 2019

Department of Computer Science & Engineering

Jyothi Hills, Panjal Road,
Vettikattiri PO, Cheruthuruthy , Thrissur,
Kerala 679531

NATIONAL ASSESSMENT AND ACCREDITATION COUNCIL

Jyothi Engineering College
NAAC Accredited College with NBA Accredited Programmes*
Approved by AICTE & affiliated to APJ Abdul Kalam Technological University
A CENTRE OF EXCELLENCE IN SCIENCE & TECHNOLOGY BY THE CATHOLIC ARCHDIOCESE OF TRICHUR
JYOTHI HILLS, VETTIKATTIRI P.O, CHERUTHURUTHY, THRISSUR. PIN-679531 PH : +91- 4884-259000, 274423 FAX : 04884-274777
NBA accredited B.Tech Programmes in Computer Science & Engineering, Electronics & Communication Engineering, Electrical & Electronics Engineering and Mechanical Engineering valid for the academic years 2016-2022. NBA accredited B.Tech Programme in Civil Engineering valid for the academic years 2019-2022.

Audio Based Drone Detection and Identification Using Deep Learning

MAIN PROJECT REPORT

AKSHAYA RAJ	**(JEC16CS006)**
ALWIN JOSEPH	**(JEC16CS014)**
ARCHANA VENUGOPAL	**(JEC16CS031)**
JACOB B ALAPPAT	**(JEC16CS053)**

in partial fulfillment for the award of the degree
of

BACHELOR OF TECHNOLOGY (B.Tech)

in

COMPUTER SCIENCE & ENGINEERING

of

A P J ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Under the guidance of

Dr. VINITH R

DECEMBER 2019

Department of Computer Science & Engineering

Department of Computer Science and Engineering

JYOTHI ENGINEERING COLLEGE, CHERUTHURUTHY

THRISSUR 679 531

DECEMBER 2019

BONAFIDE CERTIFICATE

This is to certify that the main project report entitled **Audio Based Drone Detection and Identification using Deep Learning** submitted by **Akshaya Raj (JEC16CS006), Alwin Joseph (JEC16CS014), Archana Venugopal(JEC16-CS031) and Jacob B Alappat (JEC16CS053)** in partial fulfillment of the requirements for the award of **Bachelor of Technology** degree in **Computer Science and Engineering** of **A P J Abdul Kalam Technological University** is the bonafide work carried out by them under our supervision and guidance.

Dr. Vinith R	**Mrs. Ninu Francis**	**Fr. Dr. A K George**
Project Guide	Project Coordinator	Professor &
Associate Professor	Assistant Professor	Head of The Dept
Dept. of CSE	Dept. of CSE	Dept. of CSE

JYOTHI ENGINEERING COLLEGE
LEAD KINDLY LIGHT
CREATING TECHNOLOGY LEADERS OF TOMORROW
ESTD 2002

Jyothi Engineering College
NAAC Accredited college with NBA Accredited programmes*
Approved by AICTE & Affiliated to APJ Abdul Kalam Technological University
A CENTRE OF EXCELLENCE IN SCIENCE & TECHNOLOGY BY THE CATHOLIC ARCHDIOCESE OF TRICHUR
JYOTHI HILLS, VETTIKKATTIRI P.O., CHERUTHURUTHY, THRISSUR PIN-679531 PH: +91-4884-259000, 274423 FAX : 04884-274777
www.jecc.ac.in | E-mail : info@jecc.ac.in

DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

COLLEGE VISION

Creating eminent and ethical leaders through quality professional education with emphasis on holistic excellence.

COLLEGE MISSION

- To emerge as an institution par excellence of global standards by imparting quality engineering and other professional programmes with state-of- the-art facilities.
- To equip the students with appropriate skills for a meaningful career in the global scenario.
- To inculcate ethical values among students and ignite their passion for holistic excellence through social initiatives.
- To participate in the development of society through technology incubation, entrepreneurship and industry interaction.

DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING

DEPARTMENT VISION

Creating eminent and ethical leaders in the domain of computational sciences through quality professional education with a focus on holistic learning and excellence.

DEPARTMENT MISSION

- To create technically competent and ethically conscious graduates in the field of Computer Science & Engineering by encouraging holistic learning and excellence.
- To prepare students for careers in Industry, Academia and the Government.
- To instill Entrepreneurial Orientation and research motivation among the students of the department.
- To emerge as a leader in education in the region by encouraging teaching, learning, industry and societal connect

PROGRAMME OUTCOMES (POs)

1. **Engineering Knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem Analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/Development of Solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct Investigations of Complex Problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern Tool Usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The Engineer and Society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and Sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and Team Work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project Management and Finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-Long Learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

1. The graduates shall have sound knowledge of Mathematics, Science, Engineering and Management to be able to offer practical software and hardware solutions for the problems of industry and society at large.
2. The graduates shall be able to establish themselves as practising professionals, researchers or Entrepreneurs in computer science or allied areas and shall also be able to pursue higher education in reputed institutes.
3. The graduates shall be able to communicate effectively and work in multidisciplinary teams with team spirit demonstrating value driven and ethical leadership.

PROGRAMME SPECIFIC OUTCOMES (PSOs)

1. An ability to apply knowledge of data structures and algorithms appropriate to computational problems.
2. An ability to apply knowledge of operating systems, programming languages, data management, or networking principles to computational assignments.
3. An ability to apply design, development, maintenance or evaluation of software engineering principles in the construction of computer and software systems of varying complexity and quality.
4. An ability to understand concepts involved in modeling and design of computer science applications in a way that demonstrates comprehension of the fundamentals and tradeoffs involved in design choices.

COURSE OUTCOMES (COs)

C410.1 The students will be able to analyse a current topic of professional interest and present it before an audience.

C410.2 Students will be able to identify an engineering problem, analyse it and propose a work plan to solve it.

C410.3 Students will have gained thorough knowledge in design, implementations and execution of Computer science related projects.

C410.4 Students will have attained the practical knowledge of what they learned in theory subjects.

C410.5 Students will become familiar with usage of modern tools.

C410.6 Students will have ability to plan and work in a team.

ACKNOWLEDGEMENT

We take this opportunity to express our heartfelt gratitude to all respected personalities who had guided, inspired and helped us in the successful completion of this interim project. First and foremost, we express my thanks to **The Lord Almighty** for guiding us in this endeavour and making it a success.

We take immense pleasure in thanking the **Management** of Jyothi Engineering College and **Fr. Dr. Jaison Paul Mulerikkal CMI**, principal, Jyothi Engineering College for having permitted me to carry out this seminar. Our sincere thanks to **Prof. Fr .Dr. A K George**, Head of the Department of Computer Science and Engineering for permitting us to make use of the facilities available in the department to carry out the interim project successfully.

We express our sincere gratitude to **Mrs. Ninu Francis & Mrs. Swapna B Sasi**, project coordinators for their invaluable supervision and timely suggestions. We are very happy to express our deepest gratitude to our mentor **Dr. Vinith R**, Associate Professor,Department of Computer Science and Engineering, Jyothi Engineering College for his able guidance and continuous encouragement.

Last but not least, we extend our gratefulness to all teaching and non-teaching staff who directly or indirectly involved in the successful completion of this interim project work and to all friends who have patiently extended all sorts of help for accomplishing this undertaking.

ABSTRACT

In recent years, unmanned aerial vehicles (UAVs) have become increasingly accessible to the public due to their high availability with affordable prices while being equipped with better technology. However, this raises a great concern from both the cyber and physical security perspectives since UAVs can be utilized for malicious activities in order to exploit vulnerabilities by spying on private properties, critical areas or to carry dangerous objects such as explosives which make them a great threat to the society. Drone identification is considered the first step in a multi-procedural process in securing physical infrastructure against this threat. The project aims at introducing drone detection and identification methods using deep learning technique of Convolutional Neural Network (CNN). This algorithm will be utilized to exploit the unique acoustic fingerprints of the flying drones in order to detect and identify them. The main goal of this project is to validate the usage of this methodology of drone detection and identification in real life scenarios and to provide a robust solution to the issues of drone strikes.

Keywords: Convolutional Neural Network, Drones, Unmanned Aerial Vehicle, Detection, Identification, Deep learning.

CONTENTS

List of Figures

List of Abbreviations

CNN	: *Convolutional Neural Network*
PIL	: *Plot Image Learning*
KNN	: *K Nearest Neighbour*
KWS	: *Keyword Spotting*
SDLC	: *Software Development Life Cycle*
PCL	: *Passive Coherent Location*
DCT	: *Discrete Cosine Transformation*
UAV	: *Unmanned Aerial Vehicle*
FFT	: *Fast Fourier Transformation*
RCS	: *Radar Cross Section*
LSS	: *Low altitude Slowspeed SmallRCS*
RD	: *Range Doppler*
CFAR	: *Constant False Alarm Rate*
DoA	: *Direction of Arrival*
MUSIC	: *Multiple Signal Classification*
EKF	: *Extended Kalman Filter*
UHF	: *Ultra High Frequency*

CHAPTER 1
INTRODUCTION

1.1 Overview

Drone technologies are evolving rapidly and, not surprisingly, counter-drone technologies are as well. In recent years, unmanned aerial vehicles (UAV) or drones that used to improve our daily lives with logistics and mapping support can now be programmed to destroy things remotely. What makes them lethal and effective for warfare are advancements in video-camera techniques, precision operations with improved GPS, stealth operations and faster speed. In fact, capability improvements can be seen from India's own drone procurement and manufacturing. These drones have been used to carry drugs across border, transport smuggled materials into prisons and spy over secured facilities by recording their aerial footage. Many anti-drone systems have been developed to counter these possibilities of drone-strikes. Counter-unmanned aircraft system (UAS) technologies are focusing on a multilayered defense. They're also being tasked with providing a "counter" to countermeasures. Threats from drones have become much more prevalent with the proliferation of inexpensive drones. Drones strikes are used for targeted killings by several countries. Drone identification is the first step towards securing the physical infrastructure against this threat.

This project aims at proposing a technique to implement an autonomous system to detect and identify authorized and unauthorized drones entering a restricted area using deep learning. It converts the audio into spectrogram and extract features from each block using a convolutional neural network (CNN) to increase their generality while overcoming the drawbacks of the ones using RNN or CRNN or any other existing anti-drone measures. This project also intends to exploit the importance of acoustic signature of drones for the task. It also validates the use of these deep neural network algorithms for drone detection in real life scenarios and proposes a drone detection and identification system that would be better than the existing system of drone detection and identification.

1.2 Objectives

The main objective of this project is to introduce an autonomous drone detection and identification system which can distinguish between commercial hobby drones and malicious drones based on their acoustic signatures using deep learning. It intends to demonstrate the importance of acoustic signatures for the task.

1.3 Data Description

The data for this project is taken from an open source platform known as Github and is available at https://github.com/saraalemadi/DroneAudioDataset. The complete dataset available on the website is divided into two file named Binary and Multiclass. In binary, it is further classified as drone and not drone. In multiclass, the data is further classified as Bebop, Mambo and Unknown. The data-set as a whole is generally divided into three categories: training, test and validation. As is the case with a usual deep learning problem, we would be training the model using training dataset and evaluating the performance with the test dataset.

1.4 Organization of the project

The report is organised as follow:

- **Chapter 1:Introduction** Gives an introduction to "Audio Based classification of drones using Deep Learning".
- **Chapter 2:Literature Survey** Summarizes the various existiin techniques that helps in achieving the desired result.
- **Chapter 3: Problem Statement** Discuses about the need for the proposed system
- **Chapter 4:Project Management** Contains the effective project management model to be used for the project.
- **Chapter 5:Proposed System** Describes the various steps involved to produce this project.
- **Chapter 5:System Requirements & Specification**Describes the various technologies needed for implementation.
- **Chapter 6:Conclusion** Concludes with the future scope of implementation.
- **References** Includes the references for the project.

CHAPTER 2
LITERATURE SURVEY

2.1 Different Other Anti-Drone Systems

There are several existing anti-drone systems. Few to be mentioned are RF-based [7], Radar-based, GSM Passive Coherence, etc. Other anti-drone systems have been proposed to disable the flight capability of drones and thereby combat the threats posed by such drones. One of the approaches includes shooting a net at the flying drones to physically bring them down and prevent further flight. Another approach is to shoot a laser beam at the drone to disable it. Another approach has been proposed to deceive the drone's localization system by spoong GPS.

1. **RF Based Drone Detection:** This method mainly relies on sources like drone's rotating propellers, communication, and body vibrations.

   - **Detection by analyzing the signals from Propellers**:

     Here, the drone is detected based on the signature of the signal reflected from its propellers, which could be observed by off-the-shelf wireless receiver [3]. Similar to a measuring device, a transmitter broadcasts and a receiver captures the reflected signals bounced off the drone. The reflected signal isn't continuous and its duty cycle depends on the rotation speed and size of the mechanical device, and therefore the distance between the drone and receiver [1]. For example, if the propeller rotates with the speed around 7500 to 10500 RPM (as in Bebop ARDrone), we expect to see the signature of the drone on the frequency band less than 200Hz.

   - **Detection by analyzing the communication between the drone and its controller:**

     The proposed system includes a wireless receiver that listens to the drone's communication frequency range [6]. Most of the drones sometimes communicate with their controllers ofttimes around thirty times per second to update its standing similarly on receiving the commands from the controller. The drone controller needs a higher frequency of communication to manage the drone exactly. Therefore, a system could collect wireless samples and observes the signal at a frequency band less than 100Hz, analyses them, and detects the drone's presence.

- **Detection by analyzing the vibration patters of drone's body:**

  This can be implemented either in an active or passive approach. In the active approach, the system sends out a wireless signal and observes the reected component caused by drone body vibration. In the passive approach, the wireless receiver observes the signal overheard from the drone communication and analyzes the received signal caused by the drone's body vibration. More specifically, the receiver observes the modification in reflected signal strength caused by the drone body's vibration.

2. **Radar:**

   In this method, it detects the presence of small unmanned aircraft by their radar signature, which is generated when the aircraft encounters RF pulses emitted by the detection element. These systems employ algorithms to distinguish between drones and other low-flying objects, such as birds.

3. **Combined Sensors:**

   Here, it integrates a variety of different sensor types to provide a more robust detection capability. For example, a system shall include an acoustic sensor that cues an optical camera when it detects a potential drone in the vicinity. The use of multiple detection elements can be implemented to increase the probability of successful detection, given that no individual detection method is entirely fail-proof.

4. **GSM Passive Coherence Tracking:**

   GSM-based passive coherent location (PCL) system, consists of associate degree antenna and signal process that's custom-made to the GSM wave shape and to focus on following supported multi-hypothesis following. It will use multiple GSM Base Transceiver Stations (BTS) as illuminators. the most options of this technique are coincidental reception of multiple BTS, most angular discrimination by employing a linear array, and fusion of multiple bistatic systems to boost accuracy and backbone.

2.2 Digital Television Based Passive Bistatic Radar System for Drone Detection

Due to low price and easy operation, the number of small Unmanned Aerial Vehicles (UAVs), named drones in common, has largely increased in the past few years [5]. However, these small drones are reported of being misused to perform antisocial, unsafe, and even criminal actions, such as violation of privacy, terror acts, smuggling or espionage. Drones' salient features are Low altitude, Small RCS, Slow speed (LSS), which epitomize the new detection and tracking challenge. It appears that a conventional radar detection is an effective approach that holds promise for practical application. However, conventional radar hardly meets the requirement of continuous transmission since no-stop RF radiation may raise radiation health concerns. Also, another issue that cannot be ignored is the high cost of operating a transmitter continuously. Passive radar is a potential alternative of conventional radar detection, which exploits existing TV-signal towers as transmitters of opportunity.

The main advantages of this technology are as follows: Firstly, the transmitting power of these TV-signals is strong enough for detecting drones and the signal's omnidirectional radiation makes low altitude target detection much easier. Secondly, TV-signal is free of charge, and there is no extra cost for the deployment of the dedicated transmitters. Thirdly, TV-signals are generally transmitted continuously 24/7, making it an ideal illuminator of opportunity for continuous surveillance. A typical scheme of passive radar signal processing is illustrated in the figure. After reference signal reconstruction, clutter rejection, 2D-cross correlation, and beamforming, the Range-Doppler (RD) spectrum can be derived. Then, the Constant false alarm rate (CFAR) detection method is applied. For passive bistatic radar tracking, the Direction of Arrival (DoA) estimation based on the MUltiple SIgnal Classification (MUSIC) algorithm is one of the key technologies. Finally, the tracking algorithm based on extended Kalman filter (EKF) is used to get the target trajectory. It is worth noting that longer integration time is preferred based on the Low-flying, small and slow (LSS) feature of drones and a space-time two-dimensional sliding window MUSIC algorithm is used to gain azimuth estimation accuracy.

The PBR system developed by Wuhan University is based on the software-defined radar concept. The system consists of an antenna array, analog front-end, data acquisition, and recording subsystem, and signal processing platform. The TV-signal received from the receiver, is uploaded by an optical fiber line to the computer server for data storage. The system is capable of selectively receiving UHF band TV-signal with 8MHz bandwidth, such as China Mobile Multimedia Broadcasting (CMMB) or Digital Television Terrestrial Multimedia Broadcasting (DTMB) in China. In this paper, the feasibility and validity of the digitally multichannel PBR system at Wuhan University for drone detection and tracking have been demonstrated through two typical experimental trials. It presents the practicability, and prospects of this radar system for air traffic management of low-altitude drone. So far, the system which has been applied to certain security departments has provided 24/7 monitoring, and successfully detected and tracked the misused drones.

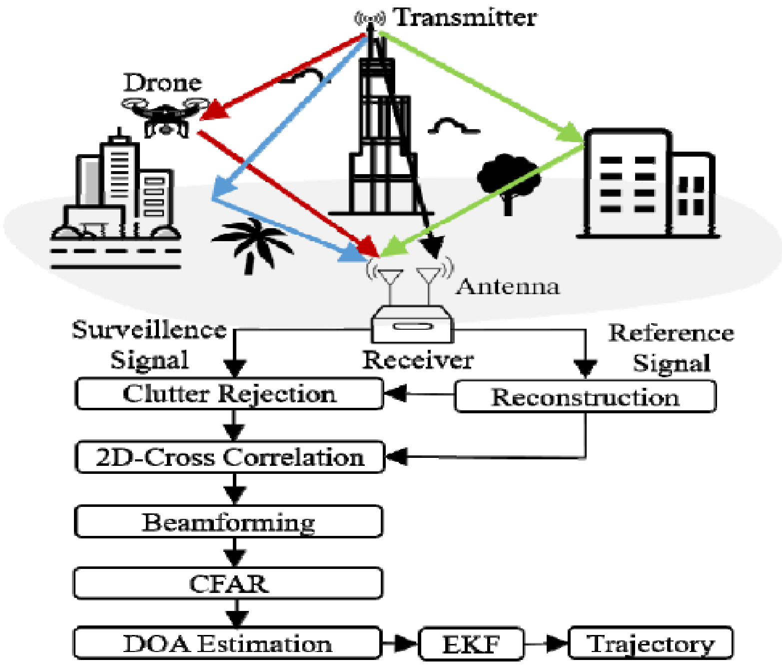

Figure 2.1: Passive Bistatic Radar Signal Processing

2.3 On the Detection of Small UAV Using a GSM Passive Coherent Location System

This paper gives an insight into the method of using GSM passive coherent system in tracking drones. In areas wherever the employment of active detection ways like (active) microwave radar doesn't seem to be attainable or unwanted, the employment of passive coherent location might play a valuable role in the detection of UAV citeknoedler2016detection. In this paper, rst the potential police work of UAV with GSM-based passive measuring device is examined. The range of little commercially accessible UAV is commonly within the vary of multiple wavelengths of the 1800 MHz GSM band (about 16 cm). It has been shown that for spherical objects, a target circumference-to-wavelength ratio of 1 or higher results in a high RCS-to-target-surface ratio (RCS, Radar Cross Section). A sure-fire detection of such of UAVs will thus be expected once applying PCL within the GSM 1800 band. subsectionPCL System Fraunhofer FKIE operates the PCL system GAMMA-2, which consists of a uniform linear array (ULA) with 16 antenna elements and a multichannel receiver. The system is capable of selectively receiving up to eight 200 kHz wide channels within a range of 30 MHz bandwidth, i.e. the system allows for the parallel reception of signals of up to eight illuminating GSM base transceiver stations (BTS). The PCL system GAMMA-2 has previously been deployed in numerous eld campaigns in coastal regions. The challenge in these eld campaigns consisted of the detection and tracking of small agile boats on the sea for surveillance tasks. The method for the evaluation of the data consists of the standard signal processing scheme used in GAMMA-2. The direct signals of the respective BTS are obtained by beamforming. To gain a ne angular grid in the surveillance signals, the antenna eld of view of 120 is divided into 64 beams. The direct signal interferences in the surveillance signals are suppressed by adaptive beamforming, and clutter cancellation. The direct signal and the surveillance signals are applied in the standard matched lter with an integration time of Tint =0 .5 s.

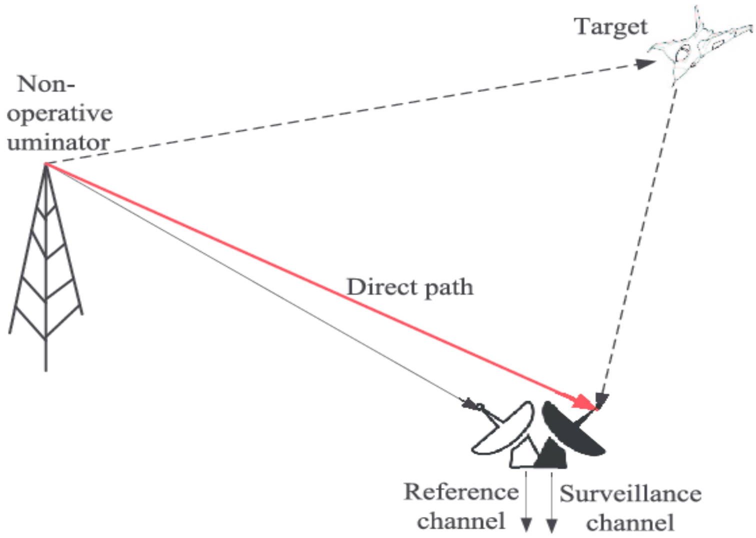

Figure 2.2: Passive Radar System

To make a decision whether the setup of the experimental trials allowed for successful detection of the drones, a visual assessment method is applied: Within each measurement and each examined Doppler frequency, the maximum peak of the range-Doppler-matrices over all examined 64 beams and overall evaluated range cells is recorded. These recorded values, depicted over time, allows for a visual impression of a possible target Doppler course. The measurement update interval was chosen to 1s. From the recorded position and velocity data of the UAV, the expected Doppler frequency of the reected signals received at the PCL system can be calculated. This ground truth Doppler course is applied as a reference for the visual evaluation of successful detection.

The detection and tracking of small UAVs are becoming an important task due to the wide availability of such devices. A rst experimental trial shows that the application of GSM based PCL systems can contribute to this challenging task. All GSM base stations used in the experiments in this paper are able to successfully detect the UAV during sections of the experiments. The complementarity of the target detection due to the multi-static case should be highlighted. The tracking of a target in GSM passive radar strongly depends on the fusion of the contribution of each base station, which constitutes a further challenging step for the successful operation of PCL based detection of UAV. In the future the fusion of passive radar with other passive sensing methods such as acoustic and optical sensors is planned to deliver an enhanced detection performance.

2.4 Real-time UAV Sound Detection and Analysis System

With the growing concern relating to the security problems with UAVs, the importance of drone(UAV) detection is being more and more stressed. Initially, as an endeavor to unravel this drawback, studies were centered on detective work drones through image knowledge [3]. First, image detection needs high-resolution cameras for prime accuracy that comes at a high worth. Second, there could also be poor visibility because of variable factors like time and weather. Whereas with sound detection, we will overcome these problems with visibility in image detection and gain a lot of stable extracted knowledge at a cheaper price.

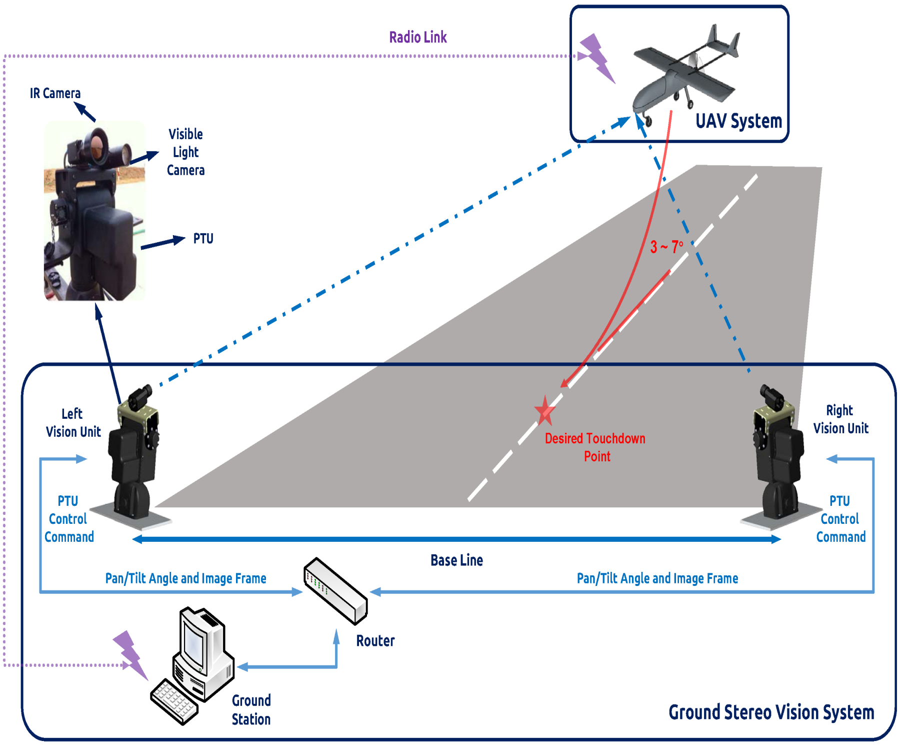

Figure 2.3: Real Time UAV System

This paper introduces a real-time drone detection and monitoring system, and the novelty of this system is that the detecting and the monitoring of the drones are available through a simple microphone [2]. This system, in operation, visualizes the audio power spectrums computed via FFT transformations of audio data. When signicant properties are shown in some of the audio data, the percentage of drone's appearance is calculated by using our machine learning and robust algorithms. The main contributions of this paper are summarized as follows:

- A Real-time detection and monitoring by advanced algorithms
- A Low-cost system using inexpensive microphones and audio digitizers.
- Unified user interface framework capable of training multiple machines learning classier for detection

Two machine learning methods are used for advanced detection algorithms utilizing those data. The rst method is PIL which resulted in 83 % accuracy. The second method is KNN which resulted in 61 percent of accuracy. These two methods contributed not only to accuracy improvement from the basic detection methods of "detecting threshold overridden amplitude", but also the increment of detecting efciency with the machine's self-learning mechanism. Without these methods, the user has to conrm each detected items without prior knowledge, whereas the machine learning methods are utilized for better detection. But the approach has some limits. First, the PIL algorithm requires large data sets of pre-stored images for accurate detection. Especially Type 2 data which are the non-drone type sounds, that are hard to collect since there are various types of sounds besides drone in public spaces. Without various and procient non-drone preset data around the system-installed environment, this algorithm leads biased result such as one-third of False (Type 2) data being determined as a "Predict True" in the test result. Second, the KNN algorithm is fast and simple, but not capable of building the hierarchies of internal representations likely necessary to support proper classication of similar, yet distinct, target. In regard to test sources, the PIL algorithm was only tested with "DJI Phantom 1" and the KNN algorithm was tested with "DJI Phantom 1,2", which makes a limitation for the generalization of these algorithms.

2.5 Speech Command Recognition with Convolutional Neural Network

The speedy development of mobile devices, interaction with machines, exploitation of voice technology has become increasing standard. Related merchandise like Google or iPhone's Siri has lead to the exploitation of speech command technology [8]. Google has also offered the service to search by voice on Android phones and a fully hands-free experience called "Ok Google" [4]. As a matter of reality, keyword spotting technology is a potential technique to provide fully hands-free interface, and this is especially convenient for mobile devices compared to typing by hands. And it's additionally the specified technique for things like driving or some emergency cases. Since the speech command recognition system sometimes runs on smartphones or tablets, it therefore must be of low-latency, and must have a very small memory footprint, that requires only very small computation. Specially for this task, the dataset is taken from Google's TensorFlow and AIY teams, which contains 65,000 WAVE audio files of people saying thirty different words. Each of the audio clips lasts for one second and contains one single word. According to completely different needs, different predefined keywords are required, such as "yes"/"no" or "up"/"down"/"left"/"right" or "stop"/"go". The predefined keywords are often reconfigurable, thus enabling the system to work for different labels with high flexibility. The system tries to classify a 1 second audio clip as either "silence", associate degree "unknown word" or one amongst the predefined keywords. Then a single-layer softmax model, a DNN model and a CNN model are used to calculate the probability that the input audio belongs to each of the labels and finally predict the label that the machine believes the input audio clip belongs to. This task is very meaningful and can be configured and run in an Android application.

CNNs are a unit higher than DNNs for Keyword Spotting (KWS) task for principally two reasons. First, DNNs simply ignore the input topology and size it into column vectors. However, for audio signals, the spectrum representations show terribly robust correlations in time and frequency. So modeling native correlations with CNNs are going to be useful and is anticipated to possess far better performance than DNNs. Second, because of the parameter sharing quality of CNNs, CNNs will have so much fewer parameters compared to DNNs for an equivalent task, therefore reducing memory footprint and procedure demand. So CNNs can have improved performance and reduced model size over DNNs and is therefore the progressive technique for KWS task.

Speech Commands Dataset from Google's TensorFlow and AIY teams are used, which consist of 65,000 WAVE audio files of people saying thirty different words, each of which lasts for one second. The data set has been separated into totally different classes like numbers, animals, directions or person names. By doing so the system can be trained with much more specific purposes. The data set is divided into three parts, including 80 percent training set, 10 percent validation set and 10 percent test set, and each subset of speech audio is classified as either silence, unknown word, or predefined keywords, that area unit hooked up totally different labels severally.

Based on human perception experiments, Mel-Frequency analysis is employed to reweight dimension of frequency and gain more perceptually-relevant representation of speech audio.

The diagram of feature exaction is shown in the Fig.

1. A 30-ms analysis window is outlined, and the speech signal is divided into totally different time frames by shifting the window (shifting stride = ten ms). Since audio signal sample is 1s each, there will (1000-30)/10+1=98 time frames.

2. After the windowing process, Fast Fourier Transformation (FFT) is calculated for each frame to obtain the frequency features, and the logarithmic Mel-Scaled filter bank is applied to the Fourier transformed frames. The last step is to calculate Discrete Cosine Transformation (DCT) to obtain the 40-dimensional coefficients vector. In this project, we have a tendency to finally obtained a [9840] 2nd matrix desired to feed the sequential neural network.

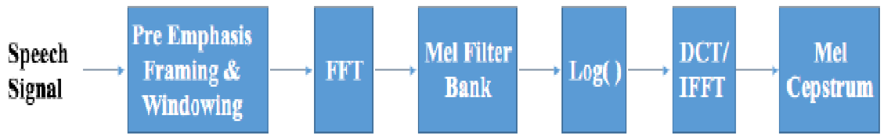

Figure 2.4: MFCC Derivation Process

A model is first built with a single hidden fully-connected layer and a softmax output layer. This simple model has just one matrix operation and bias, and therefore the range of the output nodes is same as the labels.

The second model may be a normal feed-forward absolutely connected neural network with three hidden layers and 128 hidden nodes per layer. Three hidden layers are used. A three hidden layers with a fully-connected layer neural network sometimes outperforms DNNs with one or a pair of hidden layers, however slightly worse than DNNs with 4 or more hidden layers. Another empirical expertise is to use additional hidden nodes per layer to realize higher accuracy, though it's going to cause overfitting. In this paper, dropout technique is used to prevent overfitting. For the hidden layers, there is a tendency to use corrected linear measure (ReLU) as activation functions for computing reduction, the weighted add of the output from previous layer. Compared to Vanilla Single Layer, this model is expected to give a more accurate result at the cost of more memory footprint and higher computational cost. Apart from that, DNN model is desirable for device, as its size can be easily adjusted via altering the number of parameters in the network.

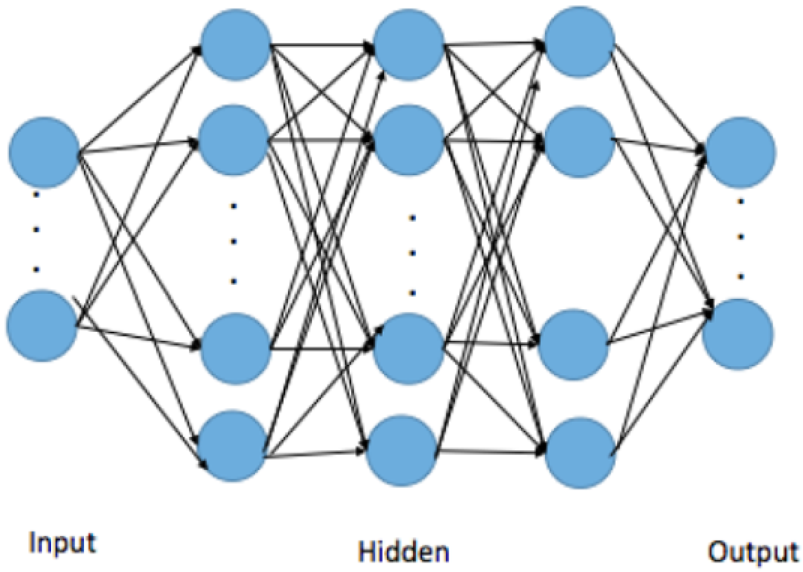

Figure 2.5: DNN Architecture

There are different key layers in CNN. They are: Convolutional (Conv) layer (multiple convolution filters to obtain different features), Pooling layer (down-sampling by taking max operation to reduce the amount of parameters and computation in the network, and hence control overfitting), Dropout layer (only keep a neuron active with some probability p, or set it to zero otherwise to control overfitting), Linear low-rank (Lin) layer (perform linear multiplication and addition to transfer the output of Conv layer to discrete nodes, reduce parameters and computation, control overfitting), and Fully-connected (FC) layer (preserve full information, or make the final softmax prediction). The CNN model is cascaded as in figure. The reason why two Conv layers are used rather than the state-of-art terribly deep and massive CNNs is to limit the quantity of parameters at the sacrifice of some accuracy.

Weights are initialized randomly from a truncated normal distribution with zero mean and specified standard deviation for symmetry breaking. Since, the ultimate worth of each weight within the trained network isn't something to be grasped, with proper data normalization it is reasonable to assume that approximately half of the weights are going to be positive and half them are going to be negative. Therefore, the weights should be very close to zero, but not identically zerobecause if every neuron in the network computes the same output, then they will also all cypher an equivalent gradients throughout back-propagation and makes the precise same parameter updates.

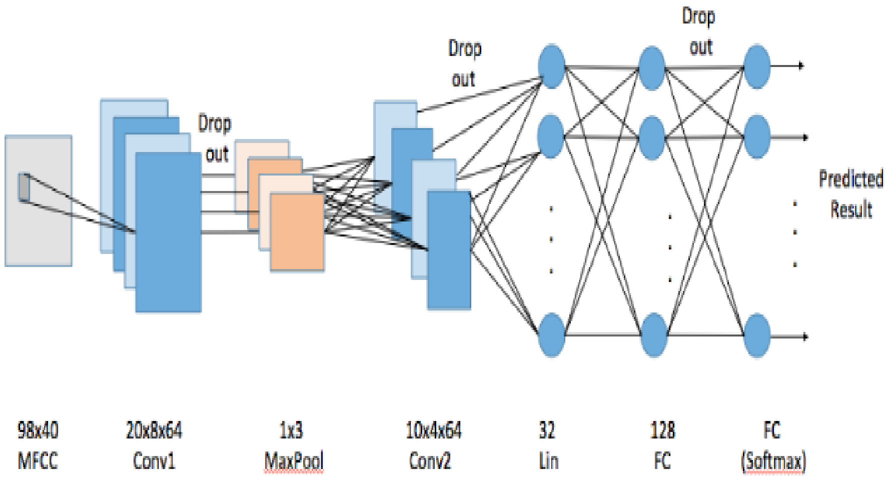

Figure 2.6: CNN Architecture

Batch size for gradient descent is one hundred. For each step, the tendency to indiscriminately select one hundred coaching samples, might break the correlation among them and build the network learn a lot of with efficiency.

A batch size which is not an adequate one is chosen, to avoid overfitting. However the batch size should not be overlarge, since coaching a neural network may be extraordinarily an intense procedure. So this price may be a trade-off between performance and hardware limitation.

The learning rate is 0.001 for the primary 5/6 of total steps followed by 0.0001 towards the top. The learning rate for the latter is comparatively smaller, since it tends to fine-tune the model for the latter steps. Many mixtures are tried and noticed that one will get each high potency and sensible convergence.

2.6 Audio Based Bird Species Identication using Deep Learning Techniques

In this paper, a brand new audio classication technique for bird species identication is proposed [9]. Whereas most approaches apply nearest neighbour matching or decision trees exploitation extracted templates for each bird species, this paper proposes the techniques from speech recognition and up to date advancements in the domain of deep learning. With novel preprocessing and data augmentation methods, the convolutional neural network is trained on the biggest publicly available dataset . The network architecture in this system achieves a mean average precision score of 0.686 when predicting the main species of each sound le and scores

0.555 when background species are used as additional prediction targets. As this performance surpasses current state of the art results, this approach won this years international BirdCLEF 2016 Recognition Challenge.

Large scale, correct bird recognition is important for vertebrate diverseness conservation. It helps us quantify the impact of land use and land management on bird species and is fundamental for bird watchers, conservation organizations, park rangers, ecology consultants, and ornithologists all over the world. Many books have been published to help humans determine the correct species and dedicated online forums exist where recordings can be shared and discussed.

The generation of good input features is vital to the success of the neural network. There are three main stages. First, the components of the sound le correspond to a bird singing/calling (signal components) and that parts contain noise or silence (noise parts) are decided. Second, the exposure for each signal and noise is figured out. Third, the spectrogram of each part is divided into equally sized chunks. Then each chunk from the signal spectrogram is used as a unique sample for training/testing and augmented with a chunk from the noise spectrogram.

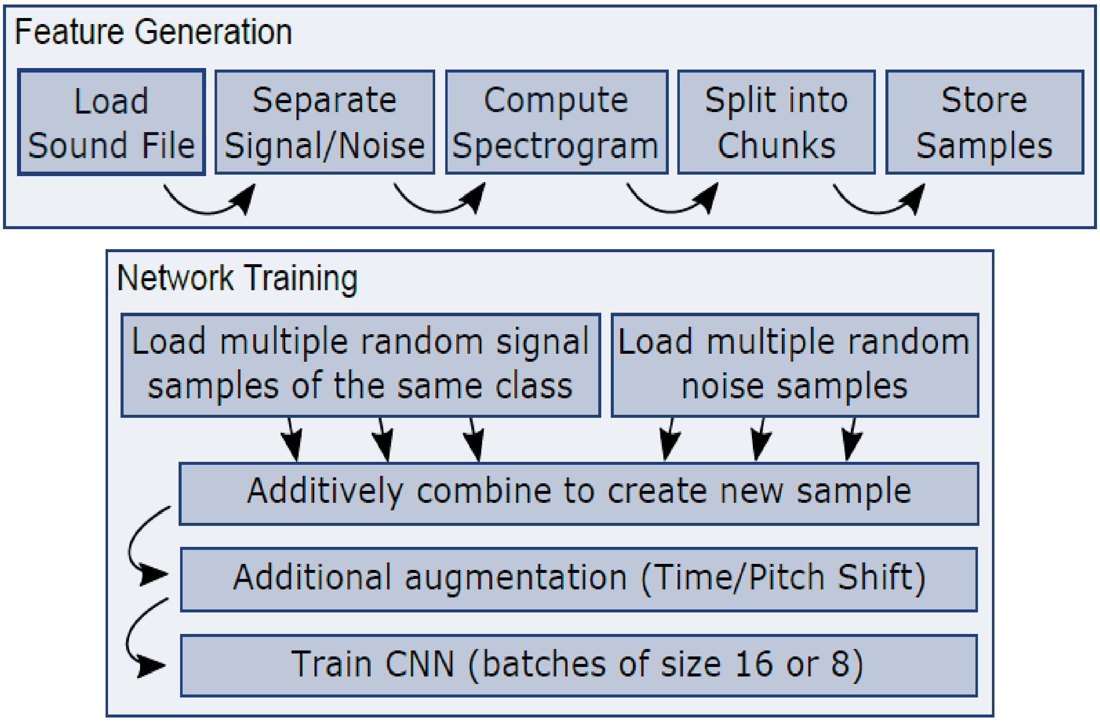

Figure 2.7: Pipeline For Training

To divide the sound le into a signal, and a noise part, rst the spectrogram of the whole le is computed. All spectrograms in this paper are computed in the same way. First the signal is passed through a short-time Fouriertransform (STFT), this is done using a Hanning window function (size 512, 75 percent overlap). Then the logarithm of the amplitude of the STFT is taken. However, the signal/noise separation is the exception to this rule because here, the logarithm of the amplitude is not taken, instead every element is divided by the maximum value, such that all values end up in the interval [0,1].

For the signal half we tend to follow quite closely. We rst select all pixels in the spectrogram that are three times bigger than the row median and three times bigger than the column median. Intuitively, this gives us all the important parts of the spectrograms, because a high amplitude usually corresponds to a bird singing/calling. We set these pixels to 1 and everything else to 0. We apply a binary erosion and dilation lter to get rid of the noise and join segments. Experimentally we found that a 4 by 4 lter produced the best results. We create a new indicator vector which has as many elements as there are columns in the spectrogram. The i-th element in this vector is set to 1 if the i-th column contains at least one 1, otherwise it is set to 0.

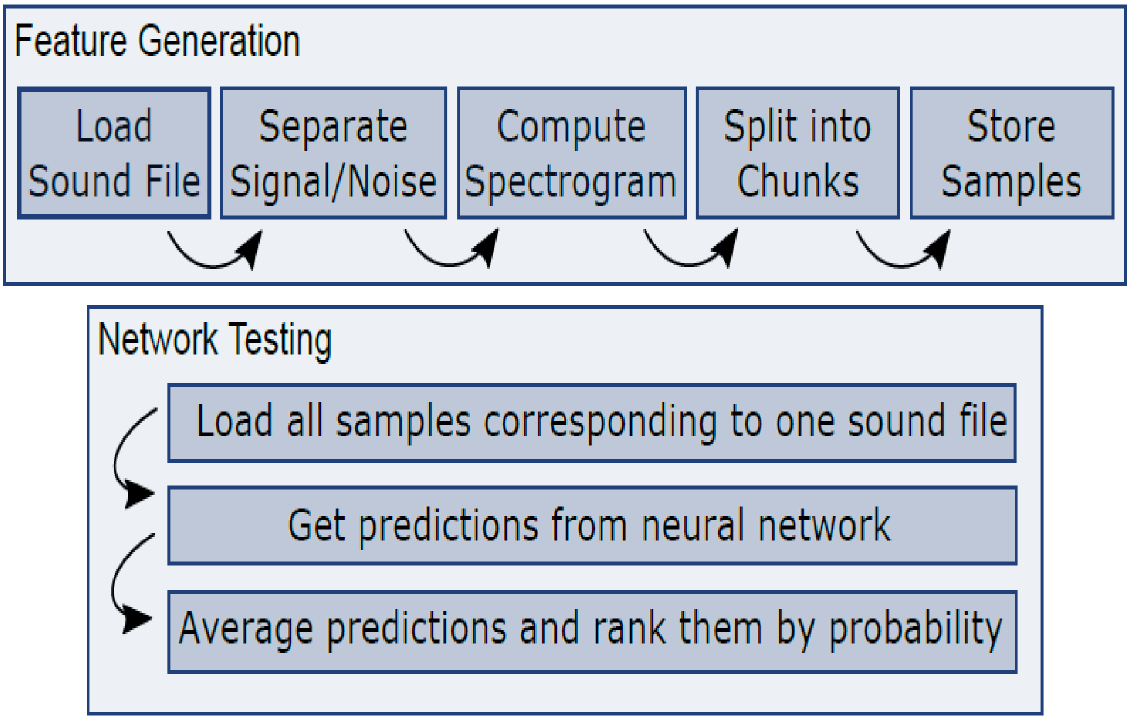

Figure 2.8: Pipeline For Testing

The indicator vector is smoothened by applying two more binary dilation lters (lter size 4 by 1). Finally the indicator vector is scaled to the length of the original sound le. It is now used as a mask to extract the signal part. For the noise part, the same steps are followed but instead of selecting the pixels which are three times bigger than row and column median, all pixels which are 2.5 times bigger than the row and column median are selected. By construction of the algorithmic program, a single column should never belong to both signal and noise part. On the other hand, it can happen that a column is not part of either noise nor signal part because we use dierent thresholds (3 versus 2.5). This is intended as it provides a safety margin for our selection process. The reasoning is that everything that was not selected as either signal nor noise, provides almost no information to the neural network. The bird is either barely audible/distorted or the sound doesn't match the conception of ground noise alright. The signal and noise masks split the sound le into many short intervals. Everything that is not selected is disregarded and not used in any future steps. The transition marks, that occur when two segments are joined together, are usually not audible because the cuts happen when no bird is calling/singing. Furthermore, the use of the dilation lters, as described earlier, ensures that the number of generated intervals are kept to a minimum value when applying the masks. From the two resulting sound les a spectrogram can be computed for both signal and noise part.

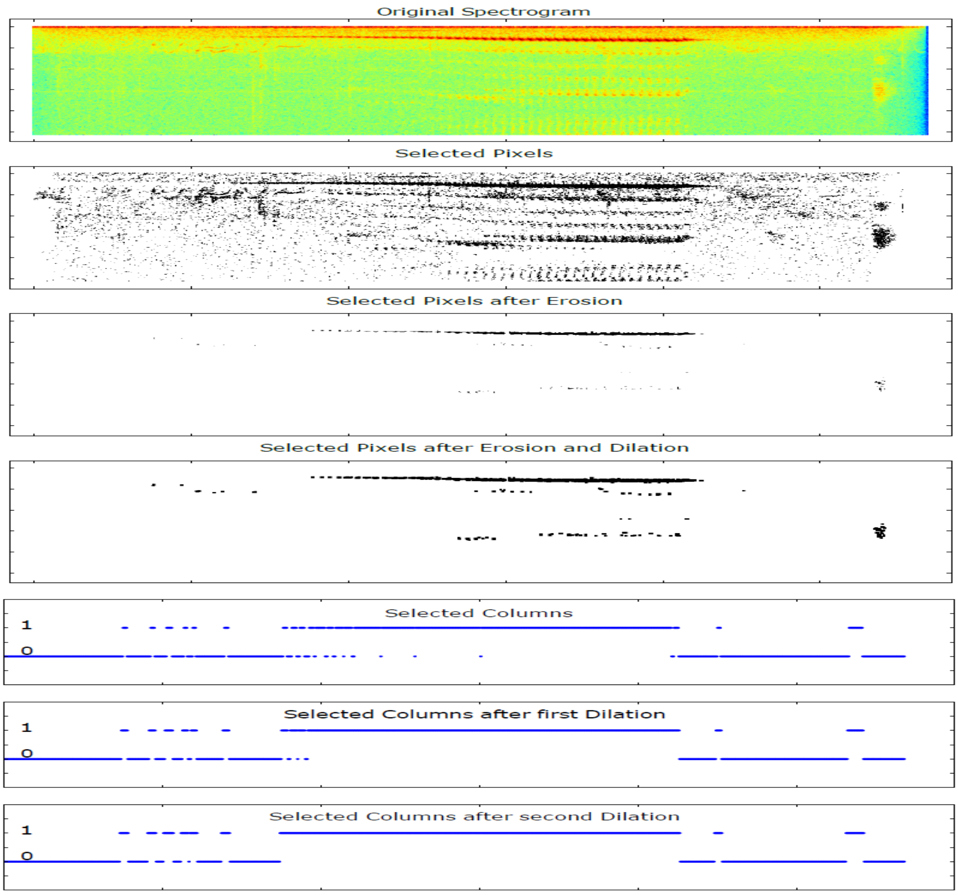

Figure 2.9: Spectrogram

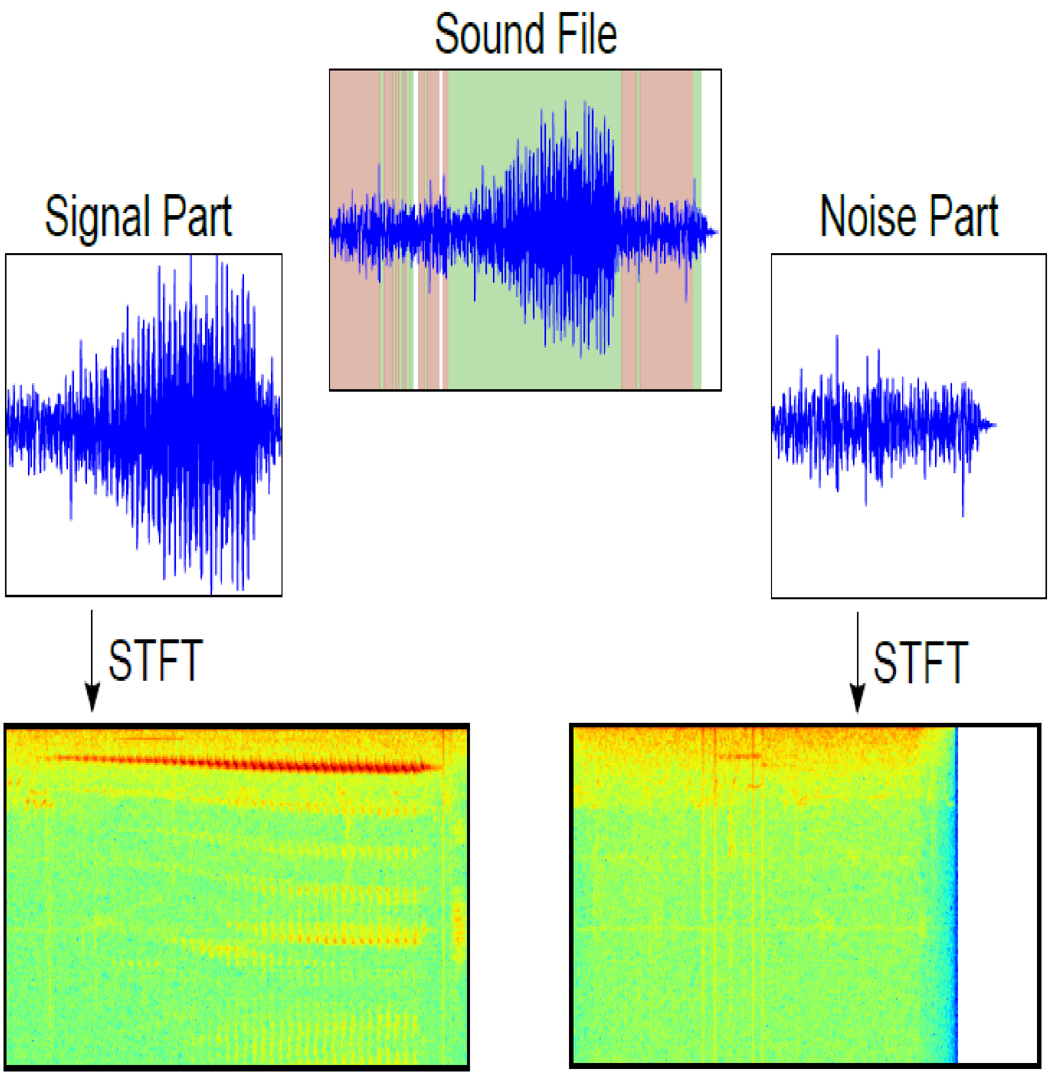

Figure 2.10: Separation of Signal and Noise part

The network contains 5 convolutional layer, each followed by a max-pooling layer. One dense layer is inserted before the nal soft-max layer. The dense layer contains 1024 and the soft-max layer 1000 units, generating a probability for each class. Batch normalization is performed before every convolutional and before the dense layer. The convolutional layers use a rectify activation function. Drop-out is used on the input layer (probability 0.2), on the dense layer (probability 0.4) and on the soft-max layer (probability 0.4). As a cost function the single label categorical cross entropy function is used (in the log domain).

Batches of 8 or 16 training examples are used. 16 training samples per batch were found to produce slightly better results but, due to memory limitations of the GPU, some models were trained with only 8 samples per batch. If many samples, from the same sound le, are present in a single batch, the performance of the batch normalization function drops considerably. Therefore, select the samples for each batch uniform at random without replacement. Normalizing the sound les beforehand might be an alternative solution.

To preserve the original label distribution, the les were grouped by their class id (species) and 10 % of each group were used for validation and the remaining 90 % for training. Training the neural network takes a lot of time. Therefore, a subset of the training set, containing 50 dierent species was chosen to ne tune parameters. This (20 times smaller) dataset enabled the testing of over 500 dierent network congurations. The nal conguration was then trained on the complete training set (considering all 999 species) and reached an accuracy score of 0.59 and a mean average precision (MAP) score of 0.67 on the local validation set (999 species). On the remote test set our best run reached a MAP score of 0.69 when considering only the main (foreground) species, 0.55 when considering the background species as well and 0.08 when only background species were considered. This means the proposed approach outperformed the next best contestant by 17% in the category where background species were ignored.

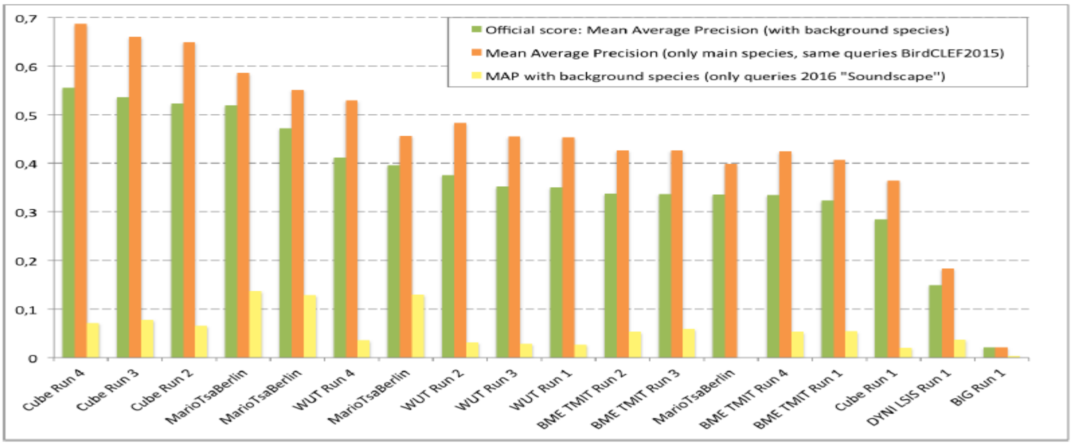

Figure 2.11: Official Scores From BirdCLEF 2016 Recognition Challenge

CHAPTER 3

PROBLEM STATEMENT

The project "Audio-based drone detection and identification using deep learning" aims at providing an efficient solution to drone classification as a counter-measure against the drone strikes happening these days. In contrast to previous, our motivation is to overcome the limitations of the existing Digital Signal Processing (DSP) based methods, radar-based methods, etc. by bringing up a more efficient and accurate classifier by using the Convolutional Neural Networks. Deep learning neural networks as we know, have their great advantages. One of the main advantages is its ability to conduct feature engineering on its own. This property distinguishes this methodology from the rest since this property helps the training model to extract all those relevant features which can lead to more accurate and precise classification. Thus, the problem of drone strikes can be identified and prevented once an efficient classifier that can overcome the limitations of the existing system is implemented.

CHAPTER 4
PROJECT MANAGEMENT

4.1 Introduction

Project management is the discipline of planning, organizing, securing, managing, leading, and controlling resources to achieve specific goals. A project is a temporary endeavor with a defined beginning and end (usually time-constrained, and often constrained by funding or deliverables), undertaken to meet unique goals and objectives, typically to bring about beneficial change or added value. The temporary nature of projects stands in contrast with business as usual (or operations), which are repetitive, permanent, or semi-permanent functional activities to produce products or services. In practice, the management of these two systems is often quite different, and as such requires the development of distinct technical skills and management strategies.

In our project we are following the typical development phases of an engineering project

1. Initiation
2. Planning and Design
3. Execution and Construction
4. Monitoring and Controlling Systems
5. Completion

4.1.1 Initiation

The initiating processes determine the nature and scope of the project. The initiating stage should include a plan that encompasses the following areas :

1. Analysing the business needs/requirements in measurable goals
2. Reviewing of the current operations
3. Financial analysis of the costs and benefits including a budget
4. Stakeholder analysis, including users, and support personal for the project

5. Project charter including costs, tasks, deliverables, and schedule

4.1.2 Planing and design

After the initiation stage, the project is planned to an appropriate level of detail (see example of a flow-chart). The main purpose is to plan time, cost and resources adequately to estimate the work needed and to effectively manage risk during project execution. As with the initiation process, a failure to adequately plan greatly reduces the project's chances of successfully accomplishing its goals.

- Determining how to plan
- Developing the scope statement
- Selecting the planning team
- Identifying deliverables and creating the work breakdown structure
- Identifying the activities needed to complete those deliverables
- Developing the schedule
- Risk planning

4.1.3 Execution

Executing consists of the processes used to complete the work defined in the project plan to accomplish the project's requirements. The execution process involves coordinating people and resources, as well as integrating and performing the activities of the project in accordance with the project management plan. The deliverables are produced as outputs from the processes performed as defined in the project management plan and other frameworks that might be applicable to the type of project at hand.

4.1.4 Monitoring & controlling

Monitoring and controlling consists of those processes performed to observe project execution so that potential problems can be identified in a timely manner and corrective action can be taken, when necessary, to control the execution of the project. The key benefit is that project performance is observed and measured regularly to identify variances from the project management plan.

4.2 System Development Life Cycle

The Systems development life cycle (SDLC), or Software development process in systems engineering, information systems, and software engineering, is a process of creating or altering information systems, and the models and methodologies that people use to develop these systems. In software engineering, the SDLC concept underpins many kinds of software development methodologies. These methodologies form the framework for planning and controlling the creation of an information system.

The SDLC phases serve as a programmatic guide to project activity and provide a flexible but consistent way to conduct projects to a depth matching the scope of the project. Each of the SDLC phase objectives is described in this section with key deliverables, a description of recommended tasks, and a summary of related control objectives for effective management. The project manager must establish and monitor control objectives during each SDLC phase while executing projects. Control objectives help to provide a clear statement of the desired result or purpose and should be used throughout the entire SDLC process.

4.2.1 Spiral Model

We have used the Spiral model in our project. The Spiral model incorporates the best characteristics of both- waterfall and prototyping model. In addition, the Spiral model also contains a new component called Risk Analysis, which is not there in the waterfall and prototype model. In the Spiral model, the basic structure of the software product is developed first. After the basic structure is developed, new features such as user interface and data administration are added to the existing software product. This functionality of the Spiral model is similar to a spiral where the circles of the spiral increase in diameter. Each circle represents a more complete version of the software product. The spiral is a risk-reduction oriented model that breaks a software project up into main projects, each addressing one or major risks. After major risks have been addressed the spiral model terminates as a waterfall model. Spiral iteration involves six steps:

1. Determine objectives, alternatives and constraints.
2. Identify and resolve risks.
3. Evaluate alternatives.
4. Develop the deliverables for the iteration and verify that they are correct.
5. Plan the next iteration.

6. Commit to an approach for the next iteration.

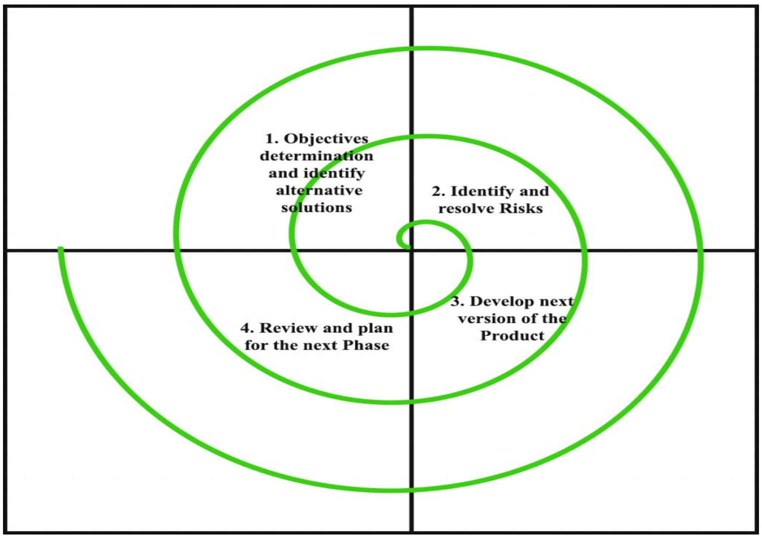

Figure 4.1: Spiral Model

CHAPTER 5
METHODOLOGY

5.1 System Requirements & Specifications

5.1.1 Spyder

Spyder is an open-source cross-platform integrated development environment (IDE) for scientific programming in the Python language. Spyder integrates with several prominent packages in the scientific Python stack, including NumPy, SciPy, Matplotlib, pandas, IPython, SymPy, and Cython, as well as other open-source software. It is released under the MIT license.

5.1.2 Windows 10

Windows 10 is a series of personal computer operating systems produced by Microsoft as part of its Windows NT family of operating systems. It is the successor to Windows 8.1 and was released to manufacturing on July 15, 2015, and to retail on July 29, 2015. Windows 10 receives new builds on an ongoing basis, which are available at no additional cost to users. Mainstream builds of Windows 10 are labeled version YYMM with YY representing the year and MM representing the month of release. For example, the latest mainstream build of Windows 10 is Version 1809. There are additional test builds of Windows 10 available to Windows Insiders. Devices in enterprise environments can receive these updates at a slower pace, or use long-term support milestones that only receive critical updates, such as security patches, over their ten-year lifespan of extended support.

5.1.3 Python 3.6.2

Python is a dynamic object-oriented programming language that can be used for many kinds of software development. It offers strong support for integration with other languages and tools, comes with extensive standard libraries, and can be learned in a few days. Many Python programmers report substantial productivity gains and feel the language encourages the development of higher quality, more maintainable code.

Python runs on Windows, Linux/Unix, Mac OS X, OS/2, Amiga, Palm Handhelds, and

Nokia mobile phones. Python has also been ported to the Java and .NET virtual machines. Python is distributed under an OSI-approved open source license that makes it free to use, even for commercial products.

5.1.4 SCIKIT-learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. The library is built upon the SciPy (Scientific Python) that must be installed before you can use scikit-learn. This stack that includes:

1. NumPy: Base n-dimensional array package
2. SciPy: Fundamental library for scientific computing
3. Matplotlib: Comprehensive 2D/3D plotting
4. IPython: Enhanced interactive console
5. Sympy: Symbolic mathematics
6. Pandas: Data structures and analysis

Extensions or modules for SciPy are conventionally named SciKits. As such, the module provides learning algorithms and is named scikit-learn.

5.1.5 Pandas

In computer programming, pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. The name is derived from the term "panel data", in econometrics term for data sets that include observations over multiple periods for the same individuals. Pandas is an open-source, BSD-licensed Python library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, statistics, analytics, etc

5.1.6 PyTorch

PyTorch is an open-source machine learning library for Python, based on Torch, used for applications such as natural language processing. It is primarily developed by Facebook's artificial-intelligence research group, and Uber's "Pyro" software for probabilistic programming is built on it. PyTorch provides two high-level features: Tensor computation (like NumPy) with strong GPU acceleration Deep Neural Networks built on a tape-based autodiff system

5.1.7 Jupyter Environment

JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data. JupyterLab is flexible: configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning. JupyterLab is extensible and modular: write plugins that add new components and integrate with existing ones.

5.2 Proposed System

Modules

5.2.1 Data Acquisition Module

More than 1300 audio clips of drone sounds were acquired. Moreover, to mimic real-life scenarios, publicly available noise datasets were used to articially augment the drone audio clips with noise. The main purpose of the articial augmentation is to measure the feasibility of the system in a noisy environment. In addition to training the learning algorithm on the augmented sound clips, a portion of the dataset is dedicated to include pure noise, silence, and drone audio clips to ensure that the system will be able to detect and identify the drone's sound from similar noises in an environment.

To acquire the drone sounds, the audio clips of the sound generated by the drone's propellers were recorded while ying and hovering in a quiet indoor environment, this will enable us to publish the dataset publicly in order to be utilized by the research community while ensuring that no privacy or security regulations are being breached.

In order to evaluate the effect of different methodological choices, we prepared three standard data sets for training and prediction:
D1: The standard training gives the model an insight into different types of scenarios where drones can be detected and identified. Different drone audio sets have been provided for this purpose.
D2:The prediction set or validation set includes the rest 30 % of the training set.
D3: The test set will be the overall leftover 20 % of the dataset, which we set aside for the final evaluation.
The forecasts will be evaluated on future data (D4 - test set).

5.2.2 Data Preprocessing Module

Deep learning is quickly becoming a powerful tool for solving complex modeling problems across a broad range of industries. An efficient model is developed through intensive training by providing a large number of datasets. These datasets contain a significant proportion of unwanted data (also called noise) in it. These data, if not removed leads to tremendous misclassification of the input data. This will ultimately degrade the performance and efficiency of the classifier. Thus, it is very important to remove unwanted noise from the dataset to improve efficiency. Therefore, the step of data preprocessing plays a very important role in contributing to the accuracy of any training model. In this project, we mainly perform three steps in the preprocessing stage. They are denoising, sampling, and segmentation.

1. **Denoising:**

   This refers to the process of removing unwanted data from the input and extracting only the required signal/data from the mixture. This forms one of the most important steps in preprocessing.

2. **Sampling:**

   A sample is defined as a smaller set of data that is chosen and/or selected from a larger population by using a predefined selection method. These sets are known as sample points, sampling units or observations. Sampling is a process used in statistical analysis in which a predetermined number of observations (samples) are taken from a larger population.

3. **Segmentation:**

   Segmentation is the process of separating the data into distinct groups. Image segmentation involves converting an image into a collection of regions of pixels that are represented by a mask or a labeled image. By dividing an image into segments, you can process only the important segments of the image instead of processing the entire image.

5.2.3 Identification and Classification Module

This module identifies and classifies the data given. Two types of classifications are used, one is the binary classification for detecting the presence of drones, and the other one is the multiclass classification purpose, used for identifying the drones. In binary classification, it classifies the given input as either drone or not drone. In multiclass classification, it classifies the input as Bebop, Mambo, or unknown.

NEURAL NETWORK IDENTIFIED

Neural network identified for classification is Convolutional Neural Network (CNN).

ALGORITHM USED: CNN

Convolutional Neural Networks or covnets are neural networks that share their parameters. Imagine an image which can be represented as a cuboid having its length, width (dimension of the image) and height (as image generally have red, green, and blue channels).

A covnet is a sequence of layers, and every layer transforms one volume to another through differentiable function.

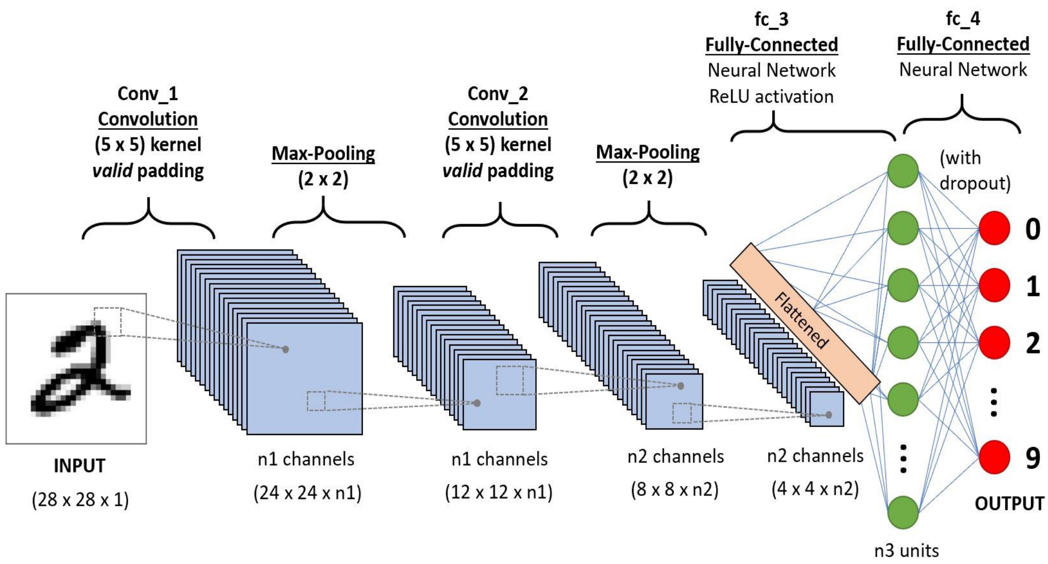

Figure 5.1: Convolutional Neural Network

Types of layers in CNN:

Consider a convnet on an image of dimension 32 x 32 x 3.

1. **Input Layer**: This layer holds the raw input of image with width 32, height 32 and depth 3.
2. **Convolution Layer**: This layer computes the output volume by computing dot product between all filters and image patch. Suppose we use total 12 filters for this layer we'll get output volume of dimension 32 x 32 x 12.
3. **Activation Function Layer**: This layer will apply element wise activation function to the output of convolution layer. Some common activation functions are RELU: max(0,

x), Sigmoid: $1/(1+e^{-x})$, Tanh, etc.

4. **Pool Layer**: This layer is periodically inserted in the covnets and its main function is to reduce the size of volume which makes the computation fast reduces memory and also prevents from overfitting. Two common types of pooling layers are max pooling and average pooling. If we use a max pool with 2 x 2 filters and stride 2, the resultant volume will be of dimension 16x16x12.

5. **Fully-Connected Layer**: This layer is regular neural network layer which takes input from the previous layer and computes the class scores and outputs the 1-D array of size equal to the number of classes.

5.3 Data Flow Diagrams

5.3.1 Data Flow Diagram- Level 0

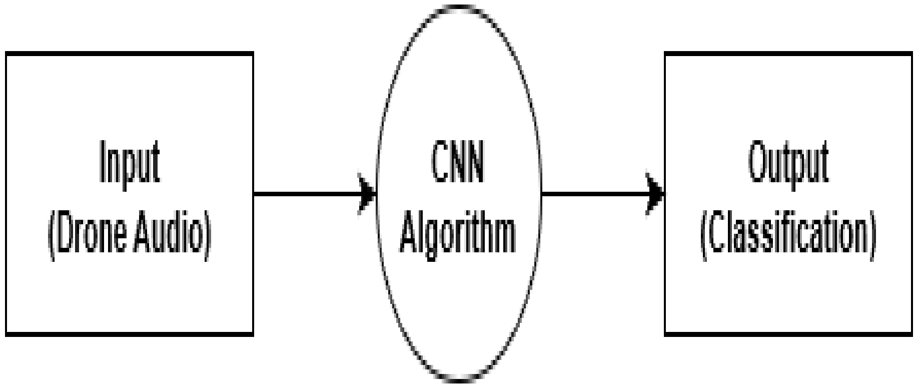

Figure 5.2: DFD- Level 0

5.3.2 Data Flow Diagram- Level 1

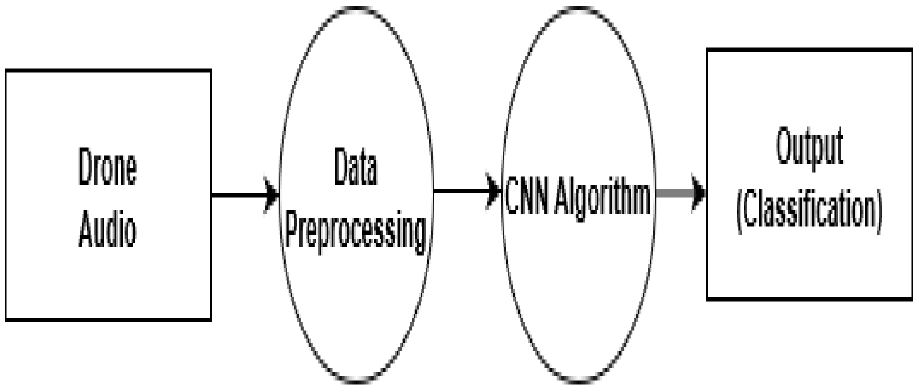

Figure 5.3: DFD- Level 1

5.3.3 Data Flow Diagram- Level 2

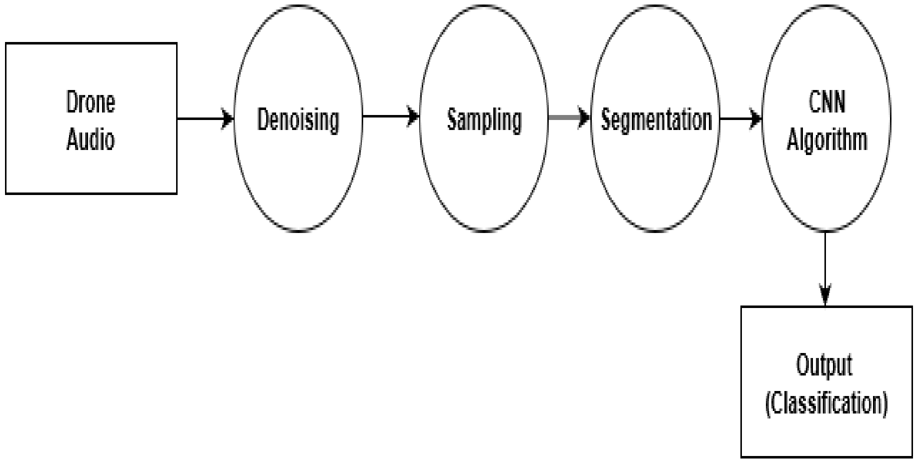

Figure 5.4: DFD- Level 2

5.4 Flow Chart

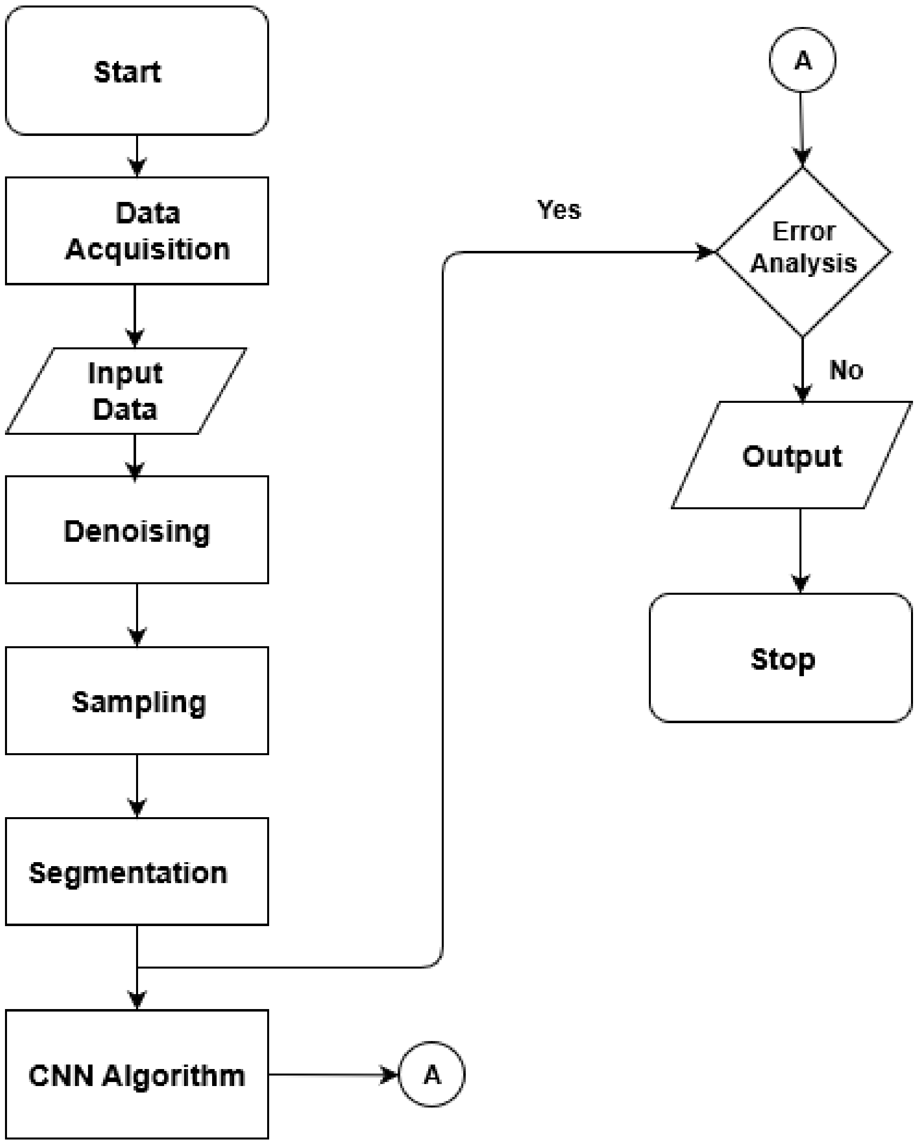

Figure 5.5: Flow Chart

5.5 Architecture

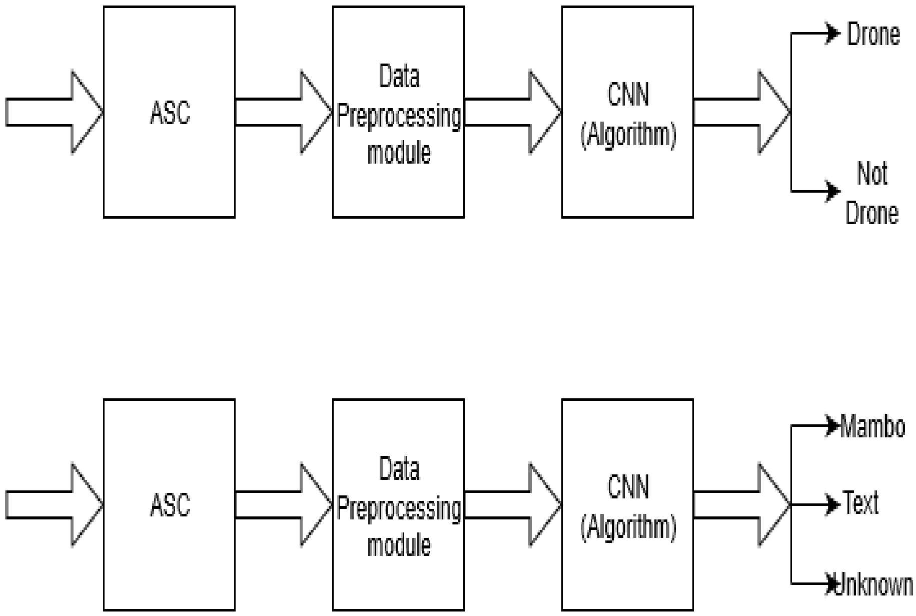

Figure 5.6: Architecture

ASC- Audio to Spectrogram Conversion

CHAPTER 6
RESULTS

In the first experiment, the effectiveness of the model in detecting drones using their acoustic signatures are examined. The evaluation metric calculated here is the accuracy of the classifier. The library used for implementing the model is Pytorch and is implemented on the Google Colab platform. For the model, an accuracy of 88.6193% was obtained.

The main goal of multiclass classification is to check the effectiveness of deep learning in drone identification. Here again, we will be using the model to identify the drone. The CNN model has been recorded with an accuracy of 88.6107% as in the above case.

CHAPTER 7

CONCLUSION AND FUTURE WORKS

Deep learning techniques have been widely used in many applications such as healthcare, big data analysis, etc. We use these deep learning techniques to detect and identify malicious drones, thus to overcome the limitations of the existing anti-drone systems and provide an efficient solution to tackle the issues caused by drone strikes.

We have outlined the design of the proposed project, which aims to identify algorithms and features that can best detect and identify the different drones entering the secured perimeters. It uses the data for two major classifications. The first one involves detection, which aims at detecting the presence of the drone. The second one aims at classifying or identifying the drone that is detected. Determining which features can best classify the drones is all done by the deep neural network by the virtue of its ability to conduct feature engineering on its own. In the future, a drone neutralization method can be included in the proposed system to extend the usage horizon of the project.

REFERENCES

[1] Andrea Bernardini, Federica Mangiatordi, Emiliano Pallotti, and Licia Capodiferro. Drone detection by acoustic signature identification. *Electronic Imaging*, 2017(10):60–64, 2017.

[2] Sungho Jeon, Jong-Woo Shin, Young-Jun Lee, Woong-Hee Kim, YoungHyoun Kwon, and Hae-Yong Yang. Empirical study of drone sound detection in real-life environment with deep neural networks. In *2017 25th European Signal Processing Conference (EUSIPCO)*, pages 1858–1862. IEEE, 2017.

[3] Juhyun Kim, Cheonbok Park, Jinwoo Ahn, Youlim Ko, Junghyun Park, and John C Gallagher. Real-time uav sound detection and analysis system. In *2017 IEEE Sensors Applications Symposium (SAS)*, pages 1–5. IEEE, 2017.

[4] Xuejiao Li and Zixuan Zhou. Speech command recognition with convolutional neural network.

[5] Yuqi Liu, Xianrong Wan, Hui Tang, Jianxin Yi, Yiyao Cheng, and Xun Zhang. Digital television based passive bistatic radar system for drone detection. In *2017 IEEE Radar Conference (RadarConf)*, pages 1493–1497. IEEE, 2017.

[6] Phuc Nguyen, Mahesh Ravindranatha, Anh Nguyen, Richard Han, and Tam Vu. Investigating cost-effective rf-based detection of drones. In *Proceedings of the 2nd Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*, pages 17–22. ACM, 2016.

[7] Phuc Nguyen, Hoang Truong, Mahesh Ravindranathan, Anh Nguyen, Richard Han, and Tam Vu. Matthan: Drone presence detection by identifying physical signatures in the drone's rf communication. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, pages 211–224. ACM, 2017.

[8] Johan Schalkwyk, Doug Beeferman, Françoise Beaufays, Bill Byrne, Ciprian Chelba, Mike Cohen, Maryam Kamvar, and Brian Strope. "your word is my command": Google search by voice: A case study. In *Advances in speech recognition*, pages 61–90. Springer, 2010.

[9] Elias Sprengel, Martin Jaggi, Yannic Kilcher, and Thomas Hofmann. Audio based bird species identification using deep learning techniques. Technical report, 2016.

[]