Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Measurements

Impulse Events Detection and Classification Software

Bachelors' thesis

Oluwaseun Olasoji

Bachelor programme: Electrical Engineering and Computer Science
Branch of study: Electronics
Supervisor: Ing. Jakub Svatoš, Ph.D.

Prague, 2023

BACHELOR‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Olasoji Oluwaseun** Personal ID number: **490526**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Measurement**

Study program: **Electrical Engineering and Computer Science**

II. Bachelor’s thesis details

Bachelor’s thesis title in English:

Impulse Events Detection and Classification Software

Bachelor’s thesis title in Czech:

Software pro detekci a klasifikaci impulsních událostí

Guidelines:

Design and implement software for the processing and classification of acoustic impulse signals. The program will be able to load data from a .wav file and further process the data. The data processing will include the algorithm for calculating cepstral coefficients (MFCC with different filter banks and GTCC). Subsequently, the implemented software will classify the recorded acoustic signals into two classes (gunshot or false alarm) using a commonly used classifier such as SVM or Neural network.

Bibliography / sources:

[1] Z. Khalilzad, Y. Kheddache, et al., An Entropy-Based Architecture for Detection of Sepsis in New-2 born Cry Diagnostic Systems, Entropy, 2022.
[2] J. Svatos, J. Holub, J. Belak, "System for an acoustic detection, localization and classification", Acta IMEKO, Vol. 10, No. 2, 2021
[3] J. Svatos, J. Holub, "Smart Acoustic Sensor", in 5th International Forum on Research and Technologies for Society and Industry. Florence, IEEE, 2019. pp. 161-165, doi: 10.1109/RTSI.2019.8895591

Name and workplace of bachelor’s thesis supervisor:

Ing. Jakub Svatoš, Ph.D. Department of Measurement FEE

Name and workplace of second bachelor’s thesis supervisor or consultant:

Date of bachelor’s thesis assignment: **15.02.2023** Deadline for bachelor thesis submission: **26.05.2023**

Assignment valid until:
by the end of summer semester 2023/2024

Ing. Jakub Svatoš, Ph.D.	Head of department’s signature	prof. Mgr. Petr Páta, Ph.D.
Supervisor’s signature		Dean’s signature

III. Assignment receipt

The student acknowledges that the bachelor’s thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor’s thesis, the author must state the names of consultants and include a list of references.

Date of assignment receipt	Student’s signature

DECLARATION

"I hereby declare that this bachelor's thesis is the product of my own independent work and that I have clearly stated all information sources used in the thesis according to Methodological Instruction No. 1/2009 – "On maintaining ethical principles when working on a university final project, CTU".
In Prague.

Date Signature

……. ……..

ACKNOWLEDGEMENTS

I would like to express my sincerest gratitude to my supervisor, Ing. Jakub Svatos for his continuous guidance and help all through the entire thesis period. This work is a product of his hard work, patience, and vision as much as it was mine.

ABSTRACT

In the world today, there is an increase in the ownership of domestic firearms. This has led to a need to be able to detect the dangerous event of a gunshot not just in military areas but in civil areas such as schools, campuses, hospitals amongst others. The current means of detection of this dangerous event today is with the use of Closed-Circuit Television (CCTV) and drone cameras which might not be completely effective in some circumstances. This means that in certain situations we would need to be able to detect, localize and classify gunshots using an automatic acoustic sensor. This would involve a lot of components that would be used together for the effective detection, and classification of the gunshot. Once an acoustic event is detected, the algorithm starts to extract features based on Mel Frequency Transformation as well as some modified versions of this transformation and some multi-label classification algorithms to confirm if the event was indeed a gunshot fired at a specific distance that can be detected. This is the basic logic of what happens within the acoustic event sensor as it can receive the sound of the surrounding environment in real time, tries to detect an acoustic event, extract some features from this event and using these features classifies it as either a gunshot or not.
The system was designed to work with recorded sound signals where we would be able to detect the interesting acoustic event and classify it. The system has been tested across different firearms with the intention of being able to detect as well as classify gunshots effectively despite environmental factors and some background noise. There were some other acoustic events which were considered such as hand claps, hand slams, door slams, bubble wraps and book slams with similar characteristics to the gunshots. The algorithm was defined to effectively classify the gunshots from the false alarms.

Keywords: gunshot, acoustic event, sensor, real time, multi-label classification, mel frequency transformation, background noise, false alarms

ABSTRAKT

V dnešním světě dochází k nárůstu vlastnictví domácích střelných zbraní. To vedlo k potřebě být schopen detekovat nebezpečnou událost výstřelu nejen ve vojenských oblastech, ale i v civilních oblastech, jako jsou mimo jiné školy, školní areály a nemocnice. Současné prostředky detekce této nebezpečné události dnes využívají uzavřený televizní okruh (CCTV) a kamery z dronů, což nemusí být za určitých okolností zcela účinné. To znamená, že v určitých situacích bychom potřebovali být schopni detekovat, lokalizovat a klasifikovat výstřely pomocí automatického akustického senzoru. To by zahrnovalo mnoho komponentů, které by byly společně použity pro účinnou detekci a klasifikaci výstřelu.
Jakmile je akustická událost detekována, algoritmus začne extrahovat rysy založené na Melově frekvenční transformaci, jakož i na některých modifikovaných verzích této transformace a některých víceznačkových klasifikačních algoritmech, aby potvrdil, zda událost byla skutečně výstřelem z určité vzdálenosti, který lze detekovat. To je základní logika toho, co se děje v rámci snímače akustických událostí, protože dokáže přijímat zvuky z okolního prostředí v reálném čase, snaží se detekovat akustickou událost, extrahovat z této události některé rysy a pomocí těchto rysů ji klasifikovat buď jako výstřel, nebo ne.
Systém byl navržen tak, aby pracoval s nahranými zvukovými signály, u nichž bychom byli schopni detekovat zajímavou akustickou událost a klasifikovat ji. Systém byl testován na různých střelných zbraních se záměrem, aby byl schopen efektivně detekovat i klasifikovat výstřely navzdory faktorům prostředí a určitému šumu v pozadí. Byly zvažovány i další akustické události, jako je tlesknutí rukou, bouchnutí rukou, bouchnutí dveřmi, zabalení bubliny a bouchnutí knihou, které mají podobné vlastnosti jako výstřely. Algoritmus byl definován tak, aby účinně klasifikoval výstřely od falešných poplachů.

Klíčová slova: výstřel, akustická událost, senzor, reálný čas, klasifikace více značek, melova frekvenční transformace, šum pozadí, falešné poplachy.

LIST OF TABLES

(Table 1) Training data distribution for SVM 33
(Table 2) Testing data distribution for SVM 33
(Table 3) 15 ms frame confusion matrix for SVM 34
(Table 4) 30 ms frame confusion matrix for SVM 35
(Table 5) 50 ms frame confusion matrix for SVM 35
(Table 6) ACC, PRC and RCC for SVM classification test 36
(Table 7) Overall ACC, PRC, RCC and MCC for SVM classification test 37
(Table 8) Training data distribution for NN 37
(Table 9) Testing data distribution for NN 38
(Table 10) 15 ms frame confusion matrix for NN 38
(Table 11) 30 ms frame confusion matrix for NN 39
(Table 12) 50 ms frame confusion matrix for NN 39
(Table 13) ACC, PRC and RCC for NN classification test 40
(Table 14) Overall ACC, PRC, RCC and MCC for NN classification test 41

LIST OF FIGURES

(Fig. 1) 7.65 mm subsonic gunshot [1] 11
(Fig. 2) Supersonic bullet [3] 12
(Fig. 3) 9 mm supersonic gunshot [1] 12
(Fig. 4) “N” wave 13
(Fig. 5) The components of a gun [5] 14
(Fig. 6) Shockwave ground reflection [3] 14
(Fig. 7) Median filter structure [7] 15
(Fig. 8) TEO Detection Block Diagram [7] 15
(Fig. 9) Correlation-based detection algorithm [7] 16
(Fig. 10) SCST block diagram [17] 17
(Fig. 11) Overview of wideband capon method [17] 18
(Fig. 12) Mel-filter bank [9] 19
(Fig. 13) Flowchart of MFCC 20
(Fig. 14) MFCC, IMFCC and LFCC filter banks [10] 20
(Fig. 15) Gammatone filter bank [11] 21
(Fig. 16) Skeleton of the feature extraction algorithms 22
(Fig. 17) Linear Classification concepts [25] 23
(Fig. 18) Maximum margin separation principle [14] 24
(Fig. 19) SVM margins [13] 24
(Fig. 20) Neural networks structure [15] 25
(Fig. 21) Logic of detection algorithm 28
(Fig. 22) Plots of gunshots 28
(Fig. 23) Plots of false alarms 29
(Fig. 24) MFCC filter bank 30
(Fig. 25) IMFCC filter bank 31
(Fig. 26) LFCC filter bank 31
(Fig. 27) GTCC filter bank 32

LIST OF ACRONYMS

AED – Acoustic Event Detector

CCTV – Closed Circuit Television

FFT – Fast Fourier Transform

LPC - Linear Prediction Coefficients

PLPC - Perceptual Linear Prediction Coefficients

ZcR - Zero-crossing Rate

SCST - Sparse Coefficient State Tracking

MFCC - Mel-frequency Cepstral Coefficients

IMFCC – Inverse Mel-frequency Cepstral Coefficients

LFCC – Linear Frequency Cepstral Coefficients

GTCC – Gammatone Cepstral Coefficients

TEO - Teager Energy Operator

AoA – Angle of Arrival

CWT - Continuous Wavelet Transform

DWT - Discrete Wavelet Transform

SVM – Support Vector Machines

ERB – Equivalent Rectangular Bandwidth

NN – Neural Networks

PRC – Precision

RCC – Recall

ACC – Accuracy

MCC – Matthews' Correlation Coefficient

CONTENTS

Acknowledgements 4
Abstract 5
List of Tables 6
List of Figures 7
List of Abbreviations 8
1 Introduction 10
2 Gunshot Analysis 11
2.1 Muzzle Blast 11
2.2 Shockwave 11
2.3 Mechanical Vibrations 14
3 Gunshot Detection 15
3.1 Median Filter Method 15
3.2 Teager Energy Operator 15
3.3 Discrete Correlation-based Algorithm 16
3.4 Sparse Coefficient State Tracking 16
3.5 Geometric Wideband Capon method 18
4 Feature Extraction 19
4.1 Mel frequency Cepstral Coefficient 19
4.2 Inverse Mel Frequency Cepstral Coefficient and Linear Frequency Cepstral Coefficient 20
4.3 Gammatone Cepstral Coefficient 21
5 Gunshot Classification 23
5.1 Support Vector Machines 23
5.2 Neural Networks 25
6 Computation 27
6.1 Computation of Gunshot Detection 27
6.2 Computation of MFCC 30
6.3 Computation of IMFCC 30
6.4 Computation of LFCC 31
6.5 Computation of GTCC 32
6.6 Computation of Classification 32
7 Implementation and Results 33
7.1 Results Using SVM Classification 33
7.2 Results Using NN Classification 37
8 Conclusion 41
8.1 Frame length 41
8.2 Feature Extraction 42
8.3 Classification 42
9 References 43

1. INTRODUCTION

There has been a recent increase in the number of public gun attacks which has led to an increase in the need for some form of protection from these attacks. The current technologies in place to provide some form of protection is in the form of surveillance technologies such as cameras and drones as well as some physical protection in the form of security guards in public places such as school campuses, hospitals, and some residential areas. The problem with these methods is that they have a limited range to which they can cover and therefore still leaves the danger of sometimes not being able to properly detect if it was a gunshot or not, sometimes not being able to tell where the sound comes from and sometimes not being able to tell if it is another dangerous event occurred (other than a gunshot). Therefore, we need to think about a system that can help detect, extract features, and classify the acoustic event effectively. This is what brought about the basic logic of acoustic surveillance systems which can help to not just detect different acoustic events within different environments but in the case of a continuous dangerous event can precisely track it to its source [6]. The major advantage of the detector is that it doesn't just detect, extract features, and classify gunshots but if strategically placed within an area can be used to monitor other acoustic events such as car crashes, dog barking, glass shattering, human screams amongst other dangerous events [1, 2].

We have had many of these acoustic surveillance devices implemented and used for military purposes in the past with the very first attempt to detect any form of acoustic event was in the First World War in Italy with special ear attachment. We have had some more military acoustic surveillance devices built such as the PILAR/PEARL and Microflown as well as some commercially used systems such as the ShotSpotter which involves an array of sensors strategically placed within a space such that it can easily locate the shooter's original location when the gunshot was initially fired [1]. The majority of the modern acoustic event detectors usually use a tetrahedron array of four microphones and have the capability to classify the gunshot to the caliber used. This tetrahedron shape helps to calculate the azimuth, elevation, and range of the gunshot. Modern systems also use a bit of artificial intelligence along with some microcontroller to read the spectrograms of the acoustic signal [18,19].

These detectors usually use numerous methods to properly detect, extract features and classify the acoustic events. In the case of a gunshot, we can first easily detect this event by the shape of the sound signal which is produced by the muzzle blast. We have seen numerous methods in the time domain to properly detect the gunshot based on the basic logic of what a muzzle blast looks like such as Absolute value method, Median filter, Teager Energy Operator (TEO), Correlation against a template Discrete Wavelet Transformation (DWT), Continuous Wavelet Transformation (CWT) amongst others [7]. This helps us to pick out the possibly dangerous acoustic events to take note of and process for further investigation. We can then take these possible dangerous acoustic events and perform some feature extraction algorithms such as Linear Prediction Coefficients (LPC), Perceptual Linear Prediction Coefficients (PLPC), Zero-crossing Rate (ZcR) or Mel-frequency Cepstral Coefficients (MFCC) with these features used for the classification [1]. Hence, the focus will be on detection, feature extraction and classification of these acoustic events.

These methods would be studied and some experimented upon. The results we get from working with these methods would be used to determine the most appropriate method to be used for Acoustic Event Detectors (AEDs) within both the time and frequency domains which can give us both very accurate results but would not require too much computational power and costs as well.

2. GUNSHOT ANALYSIS

The first thing to investigate is the possibility of being able to detect the acoustic event which in our case would be a gunshot. There are quite a few principles about gunshots, bullets, gun barrels that need to be understood for proper detection of any gunshot.

2.1. Muzzle Blast

The typical firearm uses a confined explosive charge to propel the bullet out of the gun barrel, this explosive charge causes acoustic energy coming from the center of the barrel and moves in all directions but mostly from the center itself. The explosive shock wave and the acoustic energy emanating from the barrel causes an acoustic pattern which is known as the Muzzle blast. This lasts for about 3 to 5 ms while propagating through air at the speed of sound (340 m/s) and interacts with numerous physical parameters such as temperature, humidity amongst others. There is some audio recording device with some proximity to the gunshot, if the gunshot is really close to the device, then the muzzle blast is usually the primary acoustical signal considered for proper detection. Otherwise, the recorded signal would be obscured and interfered by different barriers and obstacles which lead to different reflections and reverberations on the recorded signal. There are some handguns and rifles which produce some relatively loud sound for each gunshot fired. Therefore, to prevent this sound being detected by the audio recording device, we usually have suppressors to reduce the sound on the handguns or rifles used [1,3,4].

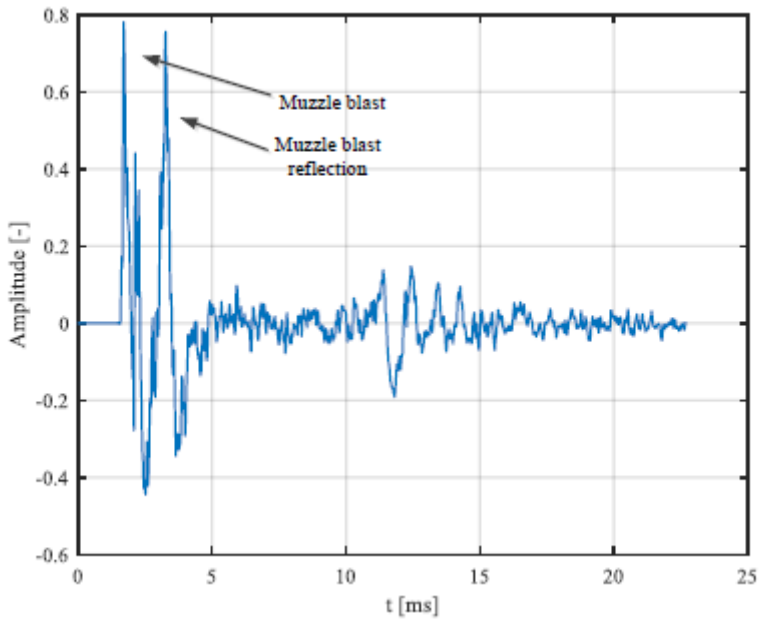

Fig 1: Recorded acoustic signal corresponding to a 7.65 mm subsonic short gun gunshot with reflection [1]

2.2. Shockwave

There is a case of the bullet moving at supersonic speed which leads to the supersonic acoustical energy moving outward from the bullet. The acoustical effect from this is known as an acoustical shockwave which expands in a conical pattern as seen below:

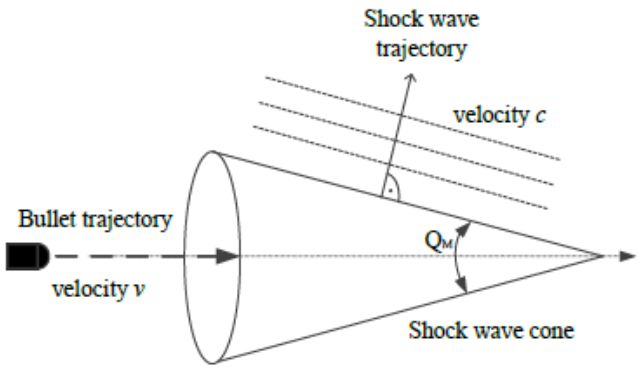

Fig 2: An acoustic wave of a supersonic bullet [3]

The bullet has an inner angle of Θ_M referred to as the Mach angle, which is dependent on the Mach number, M which is derived from the velocity of the bullet, v and the speed of sound, c as follows:

$$M = \frac{v}{c} \tag{1}$$

$$\theta_M = \arcsin\left(\frac{1}{M}\right) \tag{2}$$

We have a typical example of what the recorded shockwave would look like as shown below:

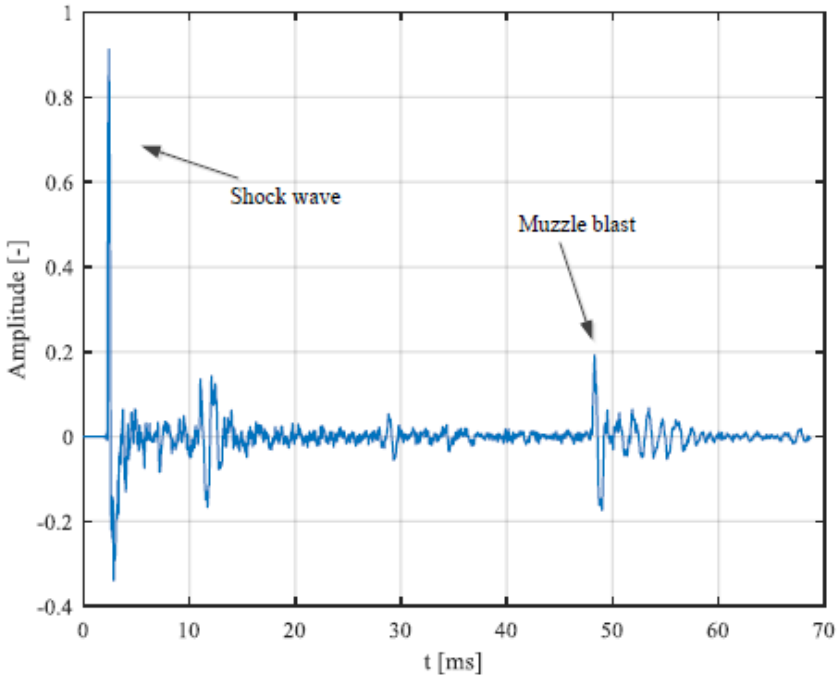

Fig 3: Recorded acoustic signal corresponding to a 9 mm supersonic short gun gunshot [1]

The acoustic shockwave has a very rapid rise to a positive over maximum pressure and then a very sudden drop to a negative under minimum pressure which creates a very distinct 'N' shape in the detected sound signal which is because of the shock wave propagating the nonlinear characteristics of air [1]. The period between the positive over maximum pressure and the negative under minimum pressure is defined as follows:

$$T \approx 1.82 \left(\frac{d}{c}\right)\left(\frac{Mx}{l}\right)^{\frac{1}{4}} \tag{3}$$

$$d = bullet\ diameter,$$
$$l = length\ of\ bullet,$$
$$c = speed\ of\ sound,$$
$$M = Mach\ number,$$
$$x = perpendicular\ distance\ between\ gun\ barrel\ and\ microphone$$

The pattern looks as is shown below with the example period being a bit less than 200μs:

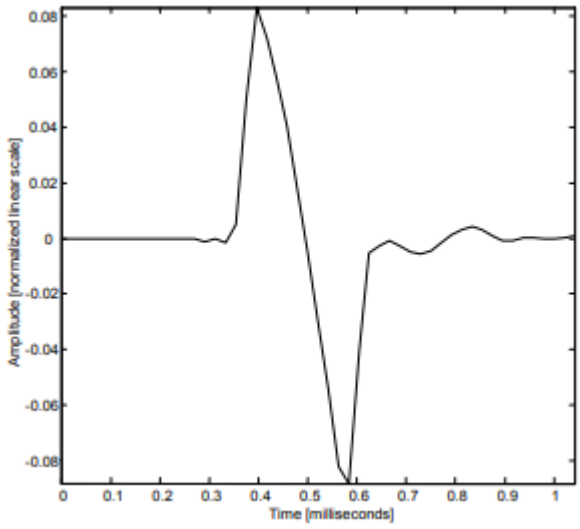

Fig 4: Shock wave recording ("N" wave)

We usually have some cases when the speed of the bullet is substantially larger than the speed of sound, this usually means that the Mach angle is small and the shockwave propagates nearly perpendicularly to the bullet's trajectory while in the case that the speed of the bullet is just slightly larger than the speed of sound, the Mach angle is almost right, and the shockwave propagates nearly parallel to the bullet's trajectory. Therefore, in the case that we have a supersonic bullet shot from a gun barrel, due to the fact of the conversion of the kinetic energy to the acoustical shockwave we can see that the Mach angle would increase and the speed decreases over time [3, 4].
Typically, the speed of sound increases with the temperature as follows:

$$c = c_0\sqrt{1 + \frac{T}{273}} \tag{4}$$

$$c = speed\ of\ sound,$$
$$c_0 = speed\ of\ sound\ at\ 0^0C = 331\frac{m}{s},$$
$$T = temperature\ in\ degree\ celsius$$

2.3. Mechanical Vibrations

In addition to the muzzle blast and the shockwave, we can usually detect a gunshot through mechanical vibrations detected by the audio recording device which could include sounds from the trigger, the hammer mechanism, the ejection of the cartridge, positioning of new ammunition by the manual or automatic system by the gun. These sounds are obviously much quieter than the muzzle blast and the shockwave which is why they are usually detected in the case that the rifle or handgun is really close to the AED.

Fig 5: The components of a gun [5]

Acoustic vibration could also be picked up by the solid surfaces around the different loud acoustic sounds. These are usually partially absorbed and partially reflected. The speed of sound is about 5 times faster in soil than it is in air so there is a short period of time before we can see the surface vibrations and the corresponding subsequent air sound signal detected by the AED. These are reflections which are usually based on the path length difference.

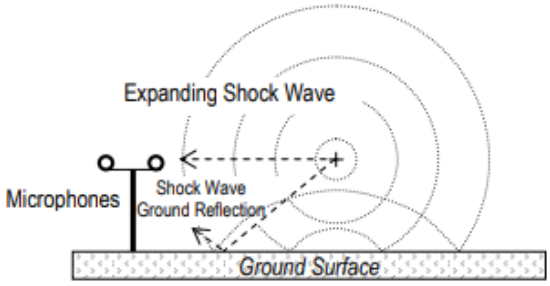

Fig 6: Shock wave ground reflection [3]

3. GUNSHOT DETECTION

There are numerous methods for gunshot detection that have been used and invented over the years. These methods are usually implemented in some form of circuit to be used in the AED. Based on different studies [7, 17, 18], there are some commonly used effective methods used that can effectively detect dangerous acoustic events. We would be looking at quite a few methods which are based on the different characteristics of gunshots explained in the previous section can easily detect the dangerous acoustic event.

3.1. Median Filter Method

We can now investigate the first method that is usually used for gunshot detection which is the Median Filter method which works on the basic principle of delay chain of taps with specific operations from the middle tap. The input signal is fed into a n-delay chain of taps with each having a delay of certain time and the input signal along with the taps are fed into a median filter whose output is subtracted from the output of the median filter. The number, *n* must be even so that we can have an odd number of inputs going into the median filter. For demonstration purposes, I would assume that we have 6 taps and delay of 1ms [7]. The basic idea of how the median filter works on is:

$$y(n) = \underset{i=0,\ldots,6}{median}\{x(n) - i\Delta_n\} \tag{5}$$

$$\Delta_n = F_s.1ms, \quad F_s = 48\ kHz$$

The block diagram of the example case would look as follows:

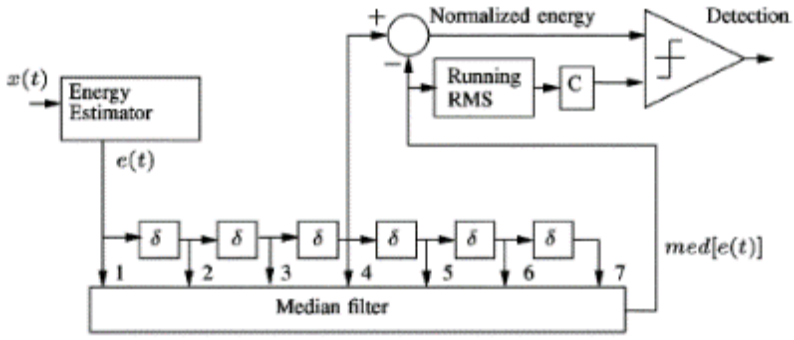

Fig 7: Block diagram of median filter structure [7]

3.2. Teager Energy Operator

We also sometimes apply a Teager Energy Operator on the estimated absolute input which has both discrete and analog forms of the operator before passing it on to the detection scheme

$$y(n) = x(n)^2 - x(n-1).x(n+1)\ [digital] \tag{6a}$$

$$y(t) = \left(\frac{dx}{dt}\right)^2 - x(t).\frac{d^2x}{dt^2}\ [analog] \tag{6b}$$

This method is said to enhance the high energy parts of the signal which helps a lot with impulsive signals. This signal is then taken in and compared with the running root mean square (RMS) value

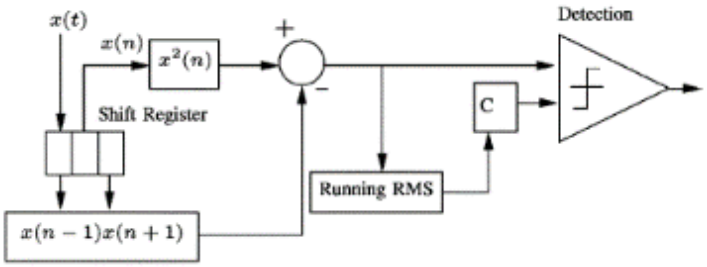

Fig 8: The TEO Detection Block Diagram [7]

3.3. DISCRETE CORRELATION-BASED ALGORITHM

Due to the different acoustical signals that could be gotten from the gunshot analysis, we can easily see that for detection purposes the use of some form of correlation might be quite useful especially when trying to detect the muzzle blast or the acoustic shockwave. The RMS helps to tell where the gunshot could be since we can tell that the higher the root mean square value the more probable a gunshot could be shot out at that time. This shows some form of correlative pattern event with the input signal itself so definitely might be able to help when trying to detect gunshots.

The basic idea of the working logic of this correlative method is that we assume that since the gunshot would be detected with a large muzzle blast if it is close to the recording device, and it would be a small muzzle blast in the case that it is still a bit far from the recording device. Hence, we would have two recording devices both at certain distances from the gunshot and we try to compute the correlation between the two input signals which then actually becomes the output which is passed to go for the running RMS value calculation as well as the possible detection. This is an important step since the correlation is a signed operation and we would not be dealing with negative numbers [7].

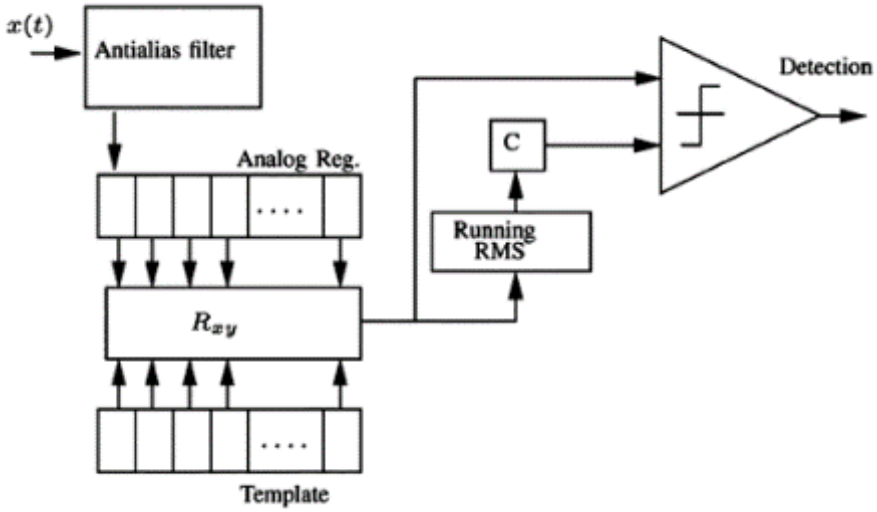

Fig 9: Basic scheme for a discrete correlation-based detection algorithm [7]

The major disadvantage of this detection algorithm is that since it is a correlative based method therefore it would be easily affected by background and environmental noises since we need the clearly defined gunshots with as much information as possible from both recording devices [7].

3.4. Sparse Coefficient State Tracking

We also have a simultaneous detection and classification algorithm known as Sparse Coefficient State Tracking (SCST). This method works by trying to separate the acoustic event from the other non-interesting events such as environmental noise, background noise amongst others. This involves two processes which are signal detection to locate the presence of a transient signal of an unknown source under the assumption that none were present and quiescent detection to find the end point of the

transient signal by searching for observations where the particularly dominant source is no longer present under the assumption that there was one present [17].
The times $\hat{e}_0$ and $\hat{e}_1$ are the estimates of the most recently observed quiescent and detection periods, respectively which for the quantization would be required to do some hypothesis tests. In the case, the data has been in quiescent period since $\hat{e}_0$ we have:

$$\begin{aligned} \mathcal{H}_0 &: z_k = w_k, \qquad \hat{e}_0 \le k \le n \\ \mathcal{H}_1^{(p)} &: z_k = \begin{cases} w_k, & \hat{e}_0 \le k \le e_1 \\ s_k^{(p)} + w_k, & e_1 \le k \le n \end{cases} \end{aligned} \tag{7a}$$

Also, when a source signal has been present since time $\hat{e}_1$, we perform the following tests:

$$\begin{aligned} \mathcal{H}_1^{(p)} &: z_k = s_k^{(p)} + w_k, \qquad \hat{e}_1 \le k \le n \\ \mathcal{H}_0 &: z_k = \begin{cases} w_k, & e_0 \le k \le n \\ s_k^{(p)} + w_k, & \hat{e}_1 \le k \le e_0 \end{cases} \end{aligned} \tag{7b}$$

$$\begin{aligned} where: \mathcal{H}_0 - null\ hypothesis, \qquad & \mathcal{H}_1^{(p)} - alternate\ hypothesis, \\ & e_0 - onset\ time\ for\ next\ unknown\ quiescent, \\ & e_1 - onset\ time\ for\ next\ source\ period, \\ & s_k^{(p)} - the\ extant\ to\ the\ unknown\ time\ e_0\ under\ the\ null\ hypothesis\ \mathcal{H}_0 \end{aligned}$$

We would have to implement the hypothesis test on streaming quantized data vectors, provide a test statistic for signal detection and evaluate the relative likelihood of the hypothesis tests, for this we use a test statistic given as:

$$B_p(n) = \max\{0, B_p(n-1) + b_p(n)\}, n = \hat{e}_0, \hat{e}_0 + 1, \dots \tag{8}$$

It is initiated by being equivalent to zero and updated by:

$$b_p(n) = \begin{cases} \ln\left(\dfrac{f_{\lambda_p}(z_n|z_{n-1})}{f_{\lambda_o}(z_n)}\right), B_p(n-1) > 0 \\ \ln\left(\dfrac{f_{\lambda_p}(z_n)}{f_{\lambda_o}(z_n)}\right), B_p(n-1) = 0 \end{cases} \tag{9}$$

$$where\ f_\lambda\ is\ the\ probability\ distribution\ of\ the\ parameter\ set\ \lambda \in \{\lambda_o, \lambda_p\}$$

Since the source is unknown, we use the following test:

$$\max_p B_p(n) \ge \eta \tag{10}$$

$$\eta - detection\ threshold\ for\ any\ source\ label$$

The moment a transient signal is detected, the quiescent detection uses the following test statistic:

$$T_p(n) = \max\{0, T_p(n-1) + t_p(n)\}, n = \hat{e}_1, \hat{e}_1 + 1, \dots \tag{11}$$

It is also initialized by being equivalent to zero and updated by:

$$t_p(n) = \ln\left(\frac{f_{\lambda_o}(z_n)}{f_{\lambda_p}(z_n|z_{n-1})}\right) \tag{12}$$

The absence of any source is defined by:

$$T_{p^*} \ge \gamma \tag{13}$$

$$where\ p^* = \ ^{\arg max}_{p} B_p(n)$$
$$\gamma - threshold\ for\ quiescent\ detection\ and\ p^*\ is\ the\ class\ label$$

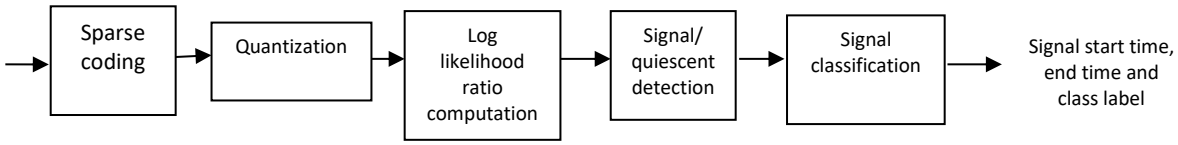

Fig 10: SCST block diagram [17].

3.5. Geometric Wideband Capon Method

There is an algorithm which helps with the detection by computing the Angle of Arrival (AoA) of the source wavefront using recorded data from different microphones in every 1-s snapshot. This method is known as the Geometric Wideband Capon method. These snapshots are partitioned into K nonoverlapping blocks of 1024 samples. We apply Fast Fourier Transform (FFT) on each block. We use these blocks to compute the sample spatial covariance matrix as follows:

$$R_{xx}(\omega_j) = \frac{1}{K}\sum_{k=1}^{K} x_k(\omega_j)\, x_k^H(\omega_j) \tag{14}$$

$$where$$
$$x_k(\omega_j) - \ the\ transformed\ vector\ for\ the\ kth\ block\ at\ narrowband$$
$$frequency\ component\ \omega_j$$

This is used to generate the geometrically averaged wideband Capon power spectrum:

$$Q_G(\theta) = \prod_{j=1}^{J} \frac{1}{v^H(\omega_j,\theta) R_{xx}^{-1}(\omega_j) v(\omega_j,\theta)} \tag{15}$$

$$where\ v(\omega_j,\theta)\ is\ the\ array\ steering\ vector\ and$$
$$\theta\ is\ the\ azimuth\ angle\ relative\ to\ the\ microphone\ array$$

The steering vector assumes the microphone inputs are ordered as: East, South, Center, West and North. The aggregated power spectrum is searched over the azimuth angle and the angles that maximize this function are the AoA angles of the detected sources of that 1-s snapshot [17].

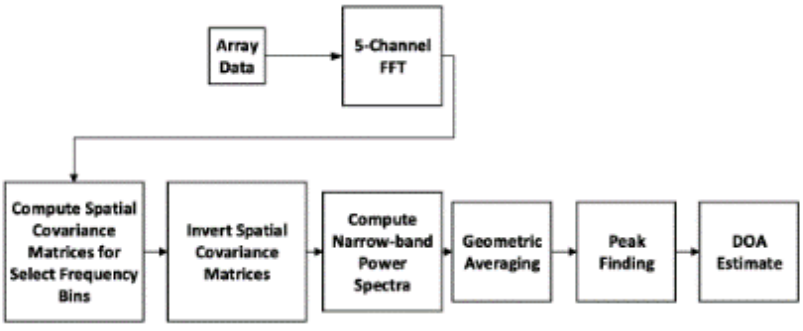

Fig. 11: Overview of wideband capon method [17].

There have been numerous other attempts to effectively detect acoustic events with optimal capacity such as with the use of Neural Networks [18] and Bayesian Networks [19]. We have also had the possibility of detecting the gunshots using Raspberry Pi with a deep learning Convolutional Neural Network classification algorithm [20]. There are other interesting methods implemented and experimented with over the years such as the use an array of microphones to locate the acoustic, even source and muzzle blast, and shock wave patterns to distinguish the gunshot using the Spatial Likelihood Function [21].

There have been quite several experimental detection algorithms which have been implemented over the years when it comes to Acoustic Events Detection with numerous advancements even with using ISR applications [22] as well as having devices which can detect devices using infrared flash of the muzzle blast and the percussion property of being able to detect from different audio sources [24]. The underlying principles all remain the same of just being able to detect the muzzle blast property or the shockwave property as described in the previous section. In this project, a customized detection algorithm based on some of the concepts from a median filter (use of RMS, Maximum threshold level) and the acoustical properties of a gunshot was implemented. This was due to the computational capabilities, ease of implementation and accuracy of the detection algorithm in relation to the others.

4. FEATURE EXTRACTION

Now that we know some dangerous acoustic events detection algorithms, we would now be trying to look at the different ways we can correctly classify the event into either gunshot or not in our case. The detection algorithm, if positive, could be a false positive which we would have to find some way to properly distinguish these false alarms from the roper gunshots

The classification involves both feature extraction and the use of some default classification algorithm. There were quite a few methods to be considered for the feature extraction as well as the classification algorithm. We have the MFCC (Mel-frequency Cepstral Coefficient), IMFCC (Inverse Mel-frequency Cepstral Coefficient), LFCC (Linear Frequency Cepstral Coefficient) and GTCC (Gammatone Cepstral Coefficient) which would be considered for the feature extraction. We would be looking at both SVM (Support Vector Machines) and Neural networks as options for the effective classification of each acoustic event detected.

4.1. Mel Frequency Cepstral Coefficient (MFCC)

The feature extraction algorithms are all based on the concept of a cepstrum. A cepstrum is the information of the rate of change in spectral bands. We usually get the periodic signals as peaks while working with them in the frequency domain by converting the input signal in the time domain via Fourier series. A non-linear rectification function (either log or power function) is applied to the peaks and then we take the spectrum of these peaks with a cosine function which is basically a Discrete Cosine Function (DCF) which results in a cepstral.[8]

Pitch is a very important concept in acoustic signals and is usually measured with frequency. Due to the fact that the human ear doesn't perceive pitch linearly, we would therefore have to be able to match the perceived frequency to the human ear frequency and therefore we would need to have a scale which could help with this matching. This scale is called the Mel Scale. The Mel scale works on the principle based on the simple fact that we know that humans can perceive the change in lower frequencies than those of higher frequencies. The Mel scale works such that we have a mapping function as follows:

$$M(f) = 1125 \log\left(1 + \frac{f}{700}\right) \tag{16}$$

The mapping function is usually derived experimentally with the following parameters:

$$M(f) \; - \; The\ mel\ frequency$$
$$f \; - \; original\ frequency$$

This Mel frequency is a psychoacoustical non-linear scale which better represent the changes in the different pitches of the human ears. Based on this scale, we can then have Mel filter banks which have a particular number of filters (between 10 and 30) which once converted are summed up together to give the Mel filter bank.

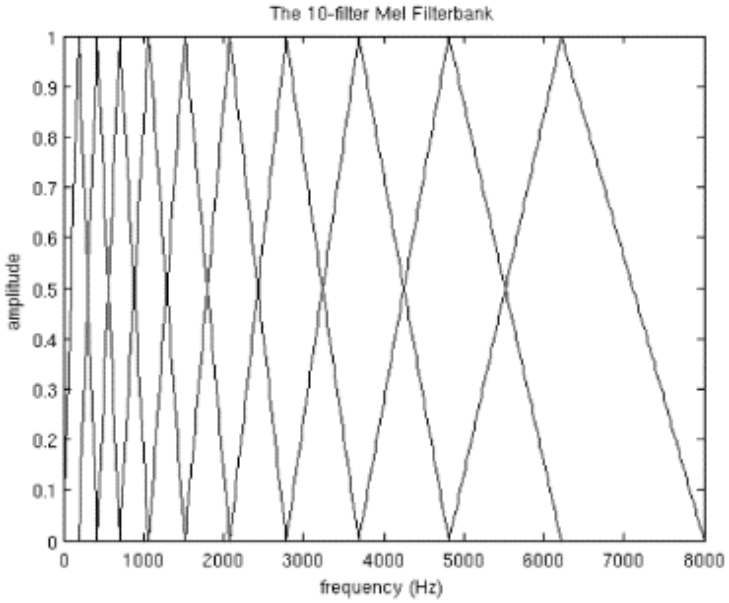

Fig 12: A Mel-filter bank containing 10 filters. This filter bank starts at 0 Hz and ends at 8000 Hz [9].

This basically defines the basic skeleton of the MFCC methodology which involves the following:

- Breaking down the input signal to overlapping time frames
- Performing some form of Fourier Transformation to these time frames (typically DFT)
- Convolution with the filter bank to produce filtered signal
- Application of some non-linear rectification function to the filtered signal (typically log10 or power function)
- Application of some form of Fourier Transformation to the rectified signal (typically DCT) which gives us the coefficients

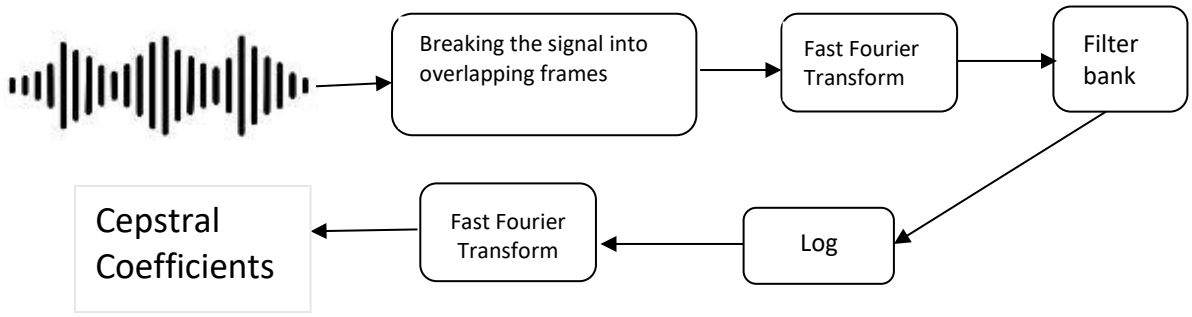

Fig 13: Flowchart of Mel-Frequency Cepstral Coefficient Feature Extraction.

4.2. Inverse Mel Frequency Cepstral Coefficient and Linear Frequency Cepstral Coefficient

We also have IMFCC and LFCC which work with the exact same methodology as shown in the flowchart above but the only difference being the filter banks used within the method. The Mel Filter banks as seen in Fig. 12 tend to have a cluster of filter banks within the lower frequencies but tend to be more spaced towards the higher frequencies while IMFCC tends to have the opposite with a lot of space for the filters within the lower frequencies and tends to get more clustered towards the higher frequencies. LFCC tends to have even spacing between all the filters within the filter banks.

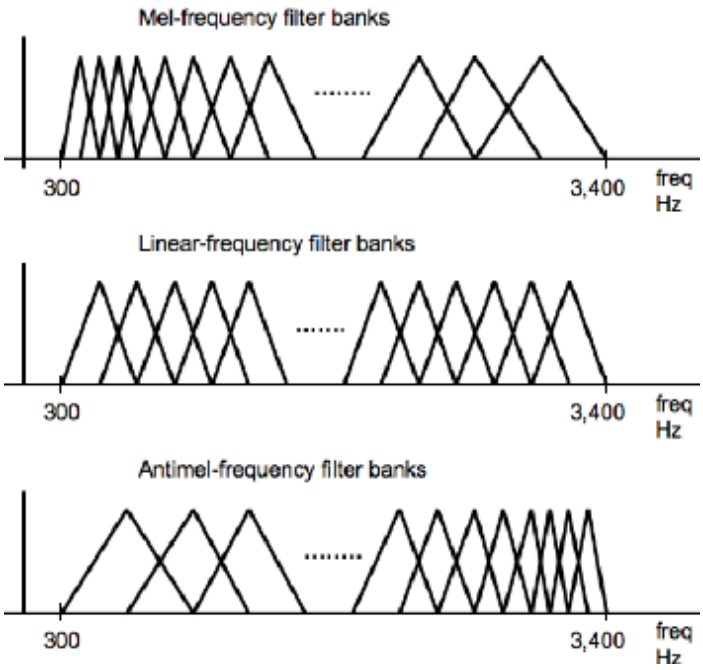

Fig. 14: The different filter banks for MFCC (top), LFCC (middle) and IMFCC (bottom).[10]

We also work with the concept of Gammatone Cepstral Coefficients which are based on the Mel filters but instead of triangular filters we use the gammatone function which is based on the human auditory response. The gammatone function is a linear function that is proportional to the filtering done by the ear which is basically a product of the gamma distribution and a sinusoidal tone. The gammatone function is given as follows:

$$g(t) = at^{n-1}e^{-2\pi bt}\cos(2\pi ft + \varphi) \quad (17)$$

$$a - amplitude,\ \ b - filter\ bandwidth\ in\ Hz,\qquad f - center\ frequency\ of\ carrier\ in\ Hz,$$
$$\varphi - phase\ of\ the\ carrier\ in\ radians,\ \ t - time\ in\ seconds$$

4.3. Gammatone Frequency Cepstral Coefficient (GTCC)

The gammatone filter bank is typically used to simulate the basilar membrane's movement with respect to time within the cochlear with the output of each filter corresponding to the frequency response of the basilar membrane within a single place. The filter bank is normally defined in such a way that the filter center frequencies are distributed across frequency in proportion to their bandwidth, known as the ERB scale. The ERB scale is approximately logarithmic, on which the filter center frequencies are equally spaced. [11]

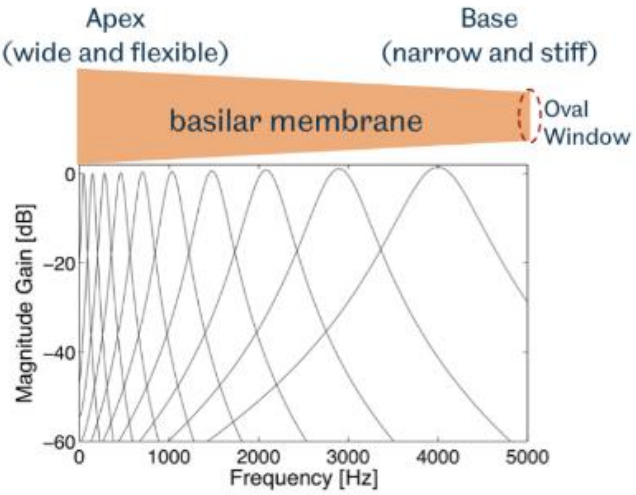

Fig 15: Gammatone filter bank [11]

From the above description, we can tell that there are numerous possibilities for the feature extraction procedure to be used for gunshot extraction purposes. The ones we would be considering have a similar skeletal principle and methodology which are applied across all four of them. The summary of the basic processes used within each possible Frequency Cepstral Coefficient extraction method can be described in the figure below, Fig 16.

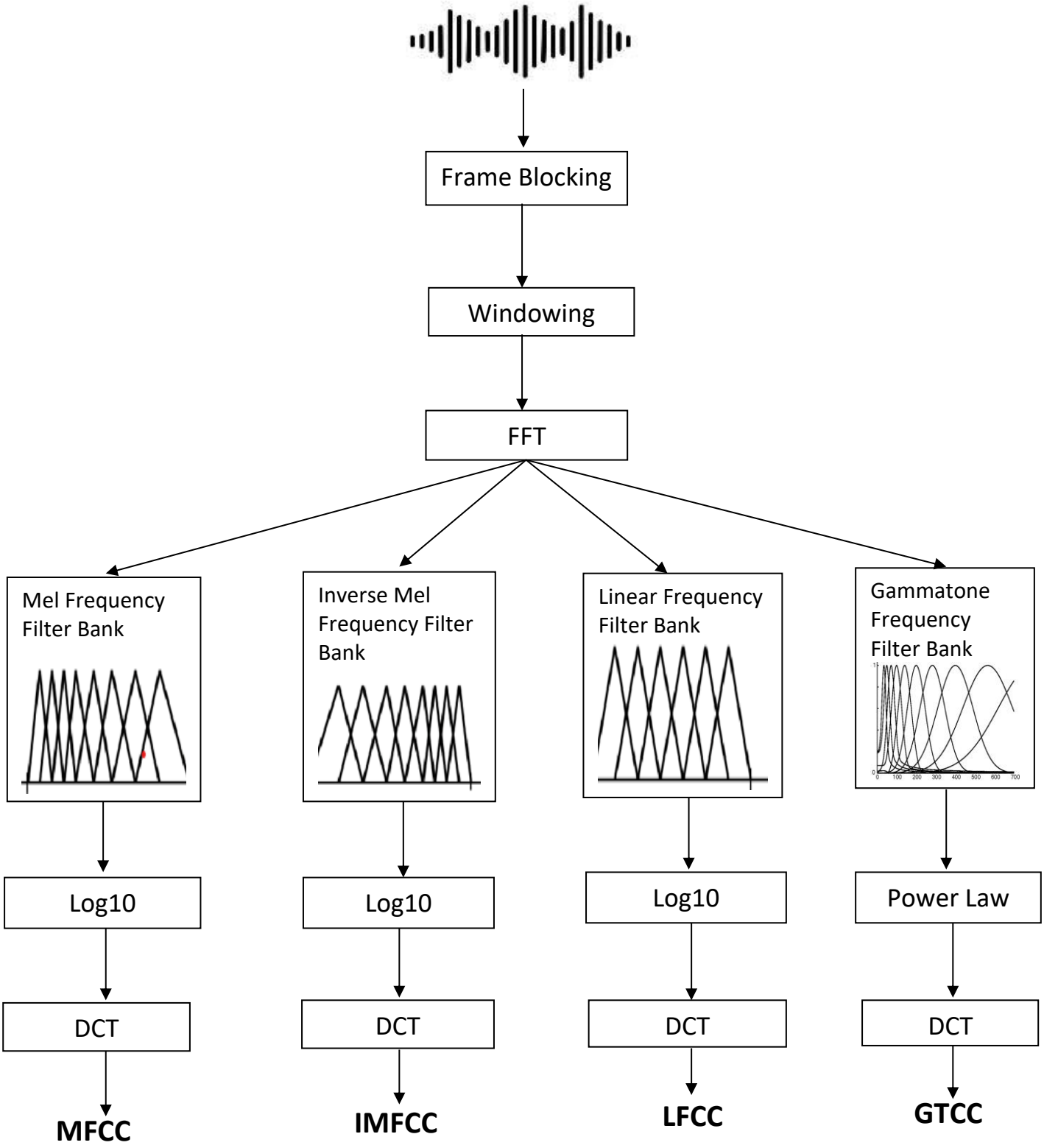

Fig. 16: The skeleton of the feature extraction algorithms

5. GUNSHOT CLASSIFICATION

Now that we have been able to categorically define the feature extraction methods which are to be considered in this case, we would start working with a way to take these extracted features and apply them into some form of classification model which could help us identify whether the acoustic event detected is a gunshot or not. We would need a very high level of accuracy for all intents and purposes. Therefore, we would be considering machine learning classification algorithms which should help produce very accurate results.

The easiest way to look at our problem would be with a two-class approach to the classification. We would have a certain number of features that we would have gotten from the feature extraction which would stand as our input data of a certain number of dimensions. Therefore, we would have a form of separating plane which can help with the classification along the different axis. There are quite a few possible methods to consider.

5.1. Support Vector Machines

One of the methods that we would consider being a two-class problem would be the Support Vector Machines (SVM). SVM works on the same principle as perceptron which is the ability to place a linear separating plane between two different classes of data but also tries to maximize the margin distance between the separating hyperplanes as well.

$$q(x) = sign(wx + b) \tag{18}$$

The basic concepts of linear classification are defined as follows:

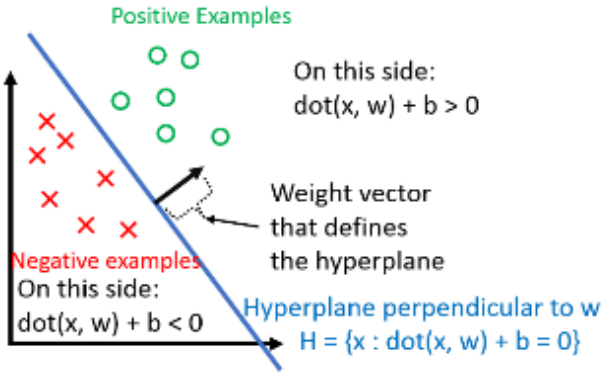

Fig 17: The basic concepts of linear classification (Perceptron) [25]

For a two-class linear classification we work with the feature vectors as follows:

$$\begin{aligned} wx + b > 0 \; when \; k = 1 \\ wx + b < 0 \; when \; k = 2 \end{aligned} \tag{19}$$

This helps define the basic separating plane which is derived from numerous feature vectors which comprises of [1 , x] with the class known to help train the model to correctly classify data to either class.

SVM, for a two-class problem, is a supervised learning method which doesn't just define the separating plane but also maximizes the margin between the two hyperplanes as follows:

$$\begin{aligned} wx + b = 1 \\ wx + b = -1 \end{aligned} \tag{20}$$

This brings about the Maximum margin problem.

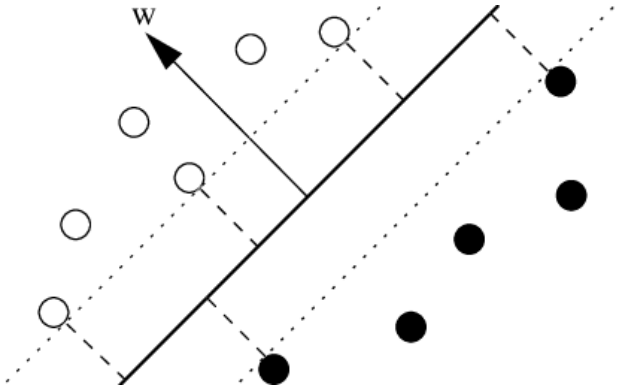

Fig 18: Image describing the maximum margin separation principle [14]

The feature vectors which have a distance from the separating plane which is about half of the margin are referred to as support vectors which are usually the closest points to the plane and help define the margin.

$$m = 2\, min_{x \in T} d(x) \tag{21}$$

$$where\ m = margin, \quad d = distance\ of\ point\ from\ separating\ plane,$$
$$x = feature\ vector, \ \ T = training\ set$$

We then try to maximize the margin by defining the signed distance of the margin from each point x which has a defined class, y (which is either 1 or -1) of the decision boundary with gradients (w, b) as follows:

$$d(x, y) = \frac{y(wx + b)}{\|w\|} \quad provided\ that\ y(wx + b) > 0 \tag{22}$$

This basically leads us to the optimization task of minimizing the maximum double the distance of the farthest point from the margin given the basic condition that its class is 1 expressed as:

$$(w^*, b^*) = \underset{w,b}{argmax} \ \ \underset{(x,y) \in T}{min} \ 2d(x, y) \ subject\ to\ y(wx + b) > 0, \forall (x, y) \in T \tag{23}$$

This leads to the margin having a value of:

$$m^* = \underset{w,b}{max} \ \ \underset{(x,y) \in T}{min} \ 2d(x, y) = \underset{w,b}{max} \frac{2}{\|w\|} \ subject\ to\ y(wx + b) > 0, \forall (x, y) \in T \tag{24}$$

We usually find it easier to minimize than to maximize due to the quadratic programming problem, so we write the expression as:

$$(w^*, b^*) = \underset{w,b}{argmin} \ \ \frac{1}{2} \|w\|^2 \ subject\ to\ y(wx + b) > 0, \forall (x, y) \in T \tag{25}$$

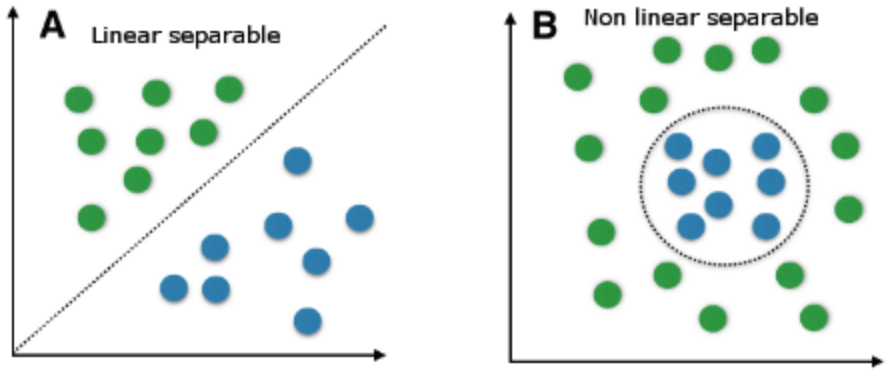

Fig 19: Image showing the different types of margins that can be gotten from SVM [13]

From the expression above, we can see that we are minimizing the gradients as well as also making sure it solves a constraint. This leads to a primal problem which we usually solve using the following expression:

$$(w^*, b^*) = \underset{(w,b)}{argmin} \left\{ \frac{1}{2}\|w\|^2 + \sum_{(x,y)\in T} f(x, y, w, b) \right\}, where \tag{26}$$

$$f(x, y, w, b) = \begin{cases} 0 & if\ y(wx + b) \geq 1 \\ \infty, & otherwise \end{cases}$$

We usually must work with Non-linear SVMs which work means the separating plane would not be a line. We approach this problem by taking the original dimension and mapping it to some higher-dimensional feature space where the training set becomes separable using some mapping function. This brings about the concept of the Kernel which relies on the inner dot product between the vectors across different dimensions. Each datapoint is mapped into high-dimensional space via some transformation:

$$\Phi: \mathrm{x} \rightarrow \varphi(\mathrm{x}) \tag{27}$$

the inner product becomes:

$$K(x_i, x_j) = \varphi(x_i).\varphi(x_j) \tag{28}$$

with K being the kernel function.

We have numerous possible kernels with the most popular ones being:

$$Linear\ kernel - K(x_i, x_j) = x_i.x_j$$
$$Polynomial\ kernel\ of\ power\ p - K(x_i, x_j) = (x_i.x_j)^p$$
$$Gaussian\ kernel - K(x_i, x_j) = e^{-\|(x_i - x_j\|^2/2\sigma^2}$$
$$Two\ layer\ perceptron - K(x_i, x_j) = \tanh(\alpha x_i.x_j + \beta)$$

5.2. Neural Networks

Another classification method that we can consider would be Neural Networks. These work by trying to replicate the basic function of a neuron in the brain which is being able to identify the different patterns and relationships between the data given input features and the output. This is usually done

by using some concepts in Statistics and Computer Science to work with, train, build and effectively test a neural network.

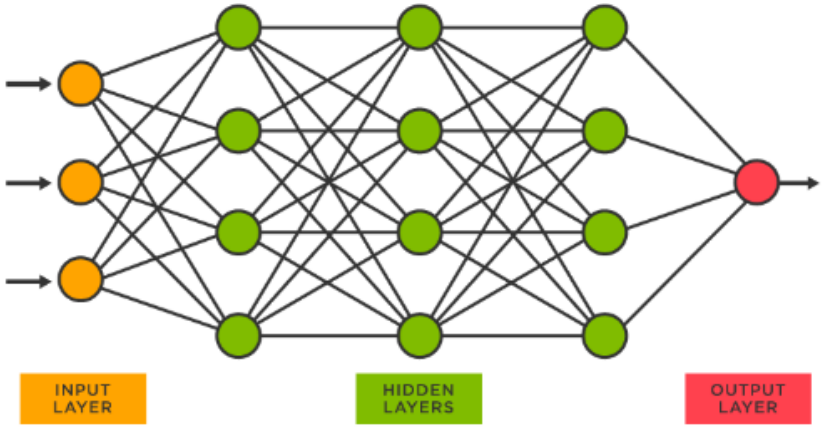

Fig 20: The basic skeleton of a neural network [15]

Neural networks typically comprise of layers of interconnected nodes which work together to define patterns within the data. There is usually an input layer, one or more hidden layers and an output layer which work with each layer having nodes with inputs of weighted sums of some nodes from the previous layer being passed on to the next layer [15].

Each node is basically a perceptron model which works with the concept of having a two-class output, y with classes, 1 and -1 given the input, x as follows:

$$y = sign(wx + b) \tag{29}$$

Therefore, we have an input layer which consists of all the input features of the data which would be trained with certain weights to be passed on to the next layer, hidden layers which take inputs from the previous hidden layer or input layer which will also be trained with those weights as well and an output layer which take in inputs from the last hidden layer and trains the weights given the final output.

An affine non-linear function is usually applied to the output at each node before passing it forward to the next layer. Some of these functions include:

$$logistic\ sigmoid\ function -\ \sigma(z) = 1/1 + e^{-z}$$
$$\tan sigmoid\ function -\ \sigma(z) = e^{z} - e^{-z}/e^{z} + e^{-z}$$
$$ReLU\ function -\ \sigma(z) = \max(0, z)$$
$$Leaky\ ReLU\ function -\ \sigma(z) = \max(0, z) + \min(0, sz)\ (0 < s < 1)$$

Usually, when working with multi label classification, we use the SoftMax function right after the output layer to get a one-hot vector.

$$[softmax(z)]_k = \frac{\exp z_k}{\sum_{l=1}^{K} \exp z_l} \tag{30}$$

The networks work with feed forwarding which as described above is just passing the input and taking weighted sums of the current layer as inputs into the next layer till we get some output. There is also the possibility of working backwards by trying to reduce the cost function by finding the weights given some predictive analysis of both inputs and outputs.[16]

6. COMPUTATION

Now, I begin to describe the parts of the thesis that was implemented in practice. The entire practical part of the thesis was done in MATLAB. There were quite a few things implemented in the detection, feature extraction and classification algorithms. We begin to describe our approach to the gunshot detection and classification problem based on the theoretical concepts studied as well as testing of these concepts as well.

6.1. Computation of Gunshot Detection

I decided to work with the basic concepts of the acoustical properties of the gunshots such as muzzle blast, shock waves and supersonic waves as well as taking some inspiration from the median filter method. Based on the above concepts, I was able to categorically define some proper conditions for which we should know that a gunshot should have on a time scale. Most of the files were audio files stored in the .wav format so it was easy to import them into MATLAB and get the sampling frequency, f_s for which would help with the signal processing. We used a continuous moving frame of different frame lengths (15 ms, 30 ms, 50 ms) of the entire gunshot with the important acoustic information required. This was done to see if it is possible to get a much better result with more acoustic information available over time or if the main muzzle blast (or shockwave) of variable length between 3ms and 6ms with a bit more information was just what was needed for proper classification. For each frame length we had some exact format to it with:

- 50 ms (10 ms before muzzle blast and 40 ms after muzzle blast)
- 30 ms (5 ms before muzzle blast with 25 ms after muzzle blast)
- 15 ms (3 ms before muzzle blast with 12 ms after muzzle blast)

The conditions were as follows:

- The maximum peak within the frame must be greater than the maximum threshold level (which is based on our knowledge of the environment which the audio was recorded).
- The maximum peak within the frame must be within the first 3 ms of the entire frame which corresponds to the muzzle blast or shock wave.
- The maximum peak within the frame must be the maximum within the entire frame of about 10ms before the start of the current frame. This was done to prevent a case of interruption of any gunshot with another gunshot from either farther away or even the reflections from the surroundings.
- The root mean square (RMS) of the frame after the muzzle blast must be greater than the root mean square of the entire audio recording. This is to make sure we are not recording a gunshot so far away from the recording device that we cannot use the information for proper classification.

$$x_{rms} = \frac{x_0}{\sqrt{2}} \tag{31}$$

$$where\ x_{rms} - root\ mean\ square\ value, x_0 - initial\ value$$

- The z-score value of the maximum peak of the frame must be the greater than the maximum threshold level defined for the environment. The definition of z-score is as follows:

$$Z = \frac{x - \mu}{\sigma} \tag{32}$$

$$where\ Z - standard\ value,$$
$$x - observed\ value,$$
$$\mu - mean\ of\ the\ sample,$$
$$\sigma - standard\ deviation\ of\ the\ sample$$

- The z-score value of the maximum peak of the frame must be the maximum z-score value of double the length of the frame. This is to make sure that we have enough time to read as much information from the acoustic signal without any interference from another gunshot and reflection. It also helps to make sure we are not measuring some one-off spiking signal but an actual high-pitched sound of a gunshot.

The algorithm was as follows:

```
Reading the necessary information and the y signal of the audio recordings
For each frame in y:
        If Max(frame) > Maximum Threshold Level
                If Max(frame) == Max(frame(1:3ms))
                        If Max(frame) > Max(frame(-10ms:0))
                                If rms(frame(4ms: frame length)) > rms(y)
                                        If z-score(frame) > Maximum Threshold Level
                                                If z-score(frame) > z-score(y(start of frame:2*frame length))
                                                        -(Save gunshot information)
                                                        -(Move the frame by about 2 times the frame length)

                                                                ……..
Else:
        Move the frame by about 1ms
```

Using this algorithm, I was able to detect some gunshots of different calibers (9mm, 5.56mm) as well as properly detect some false alarms (bubble wrap, book slam, door slam, hand slam, hand clap).

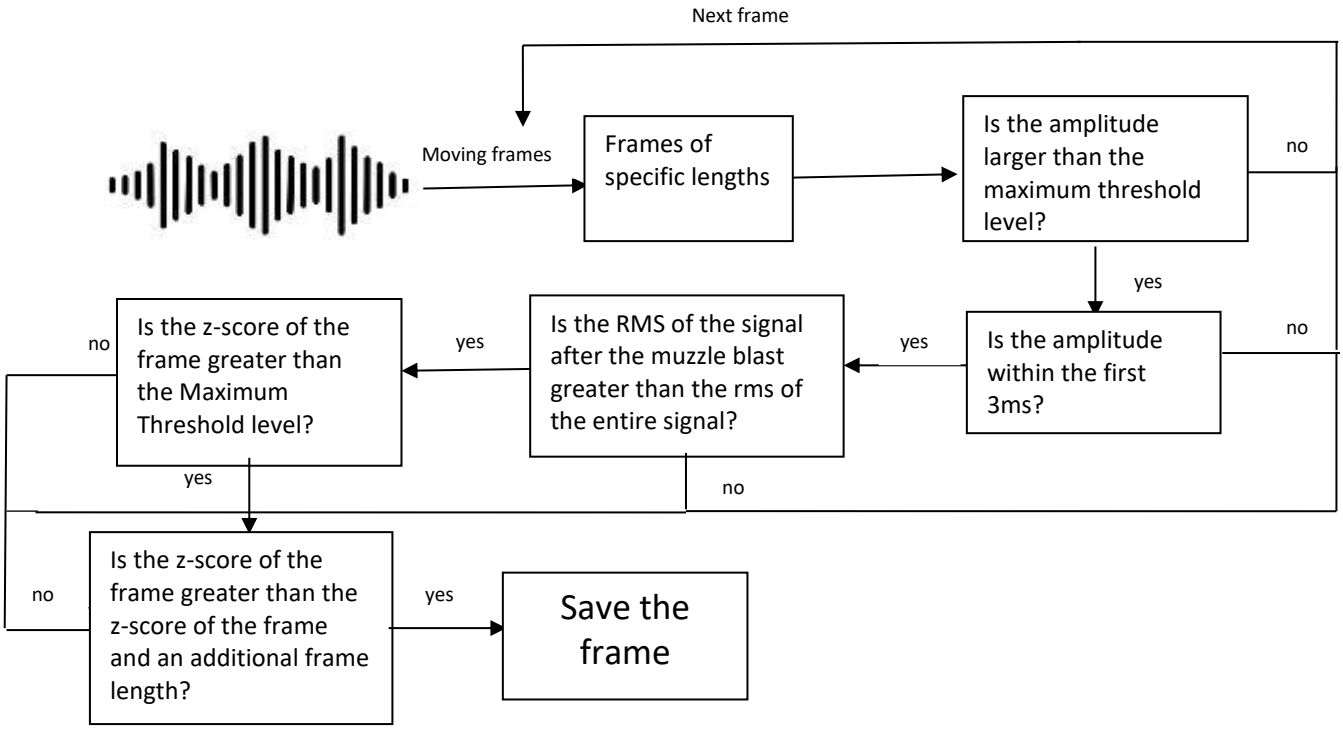

Fig. 21: The logic diagram of the detection algorithm used

The detection algorithm was applied across different firearms and false alarms which led to some frames being picked for such characteristics as seen below:

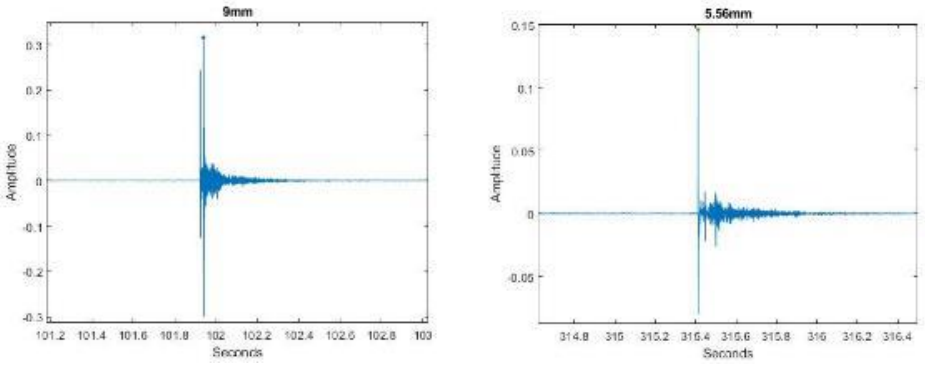

Fig 22: Plots showing detected gunshots of certain calibers (9 mm and 5.56 mm)

From the plots above, it is very clear how similar the two signals from the two calibers are on the time-based signal. We can also clearly see that the false alarms have similar properties to the gunshots which could pose a problem. Hence the need for proper feature extraction.

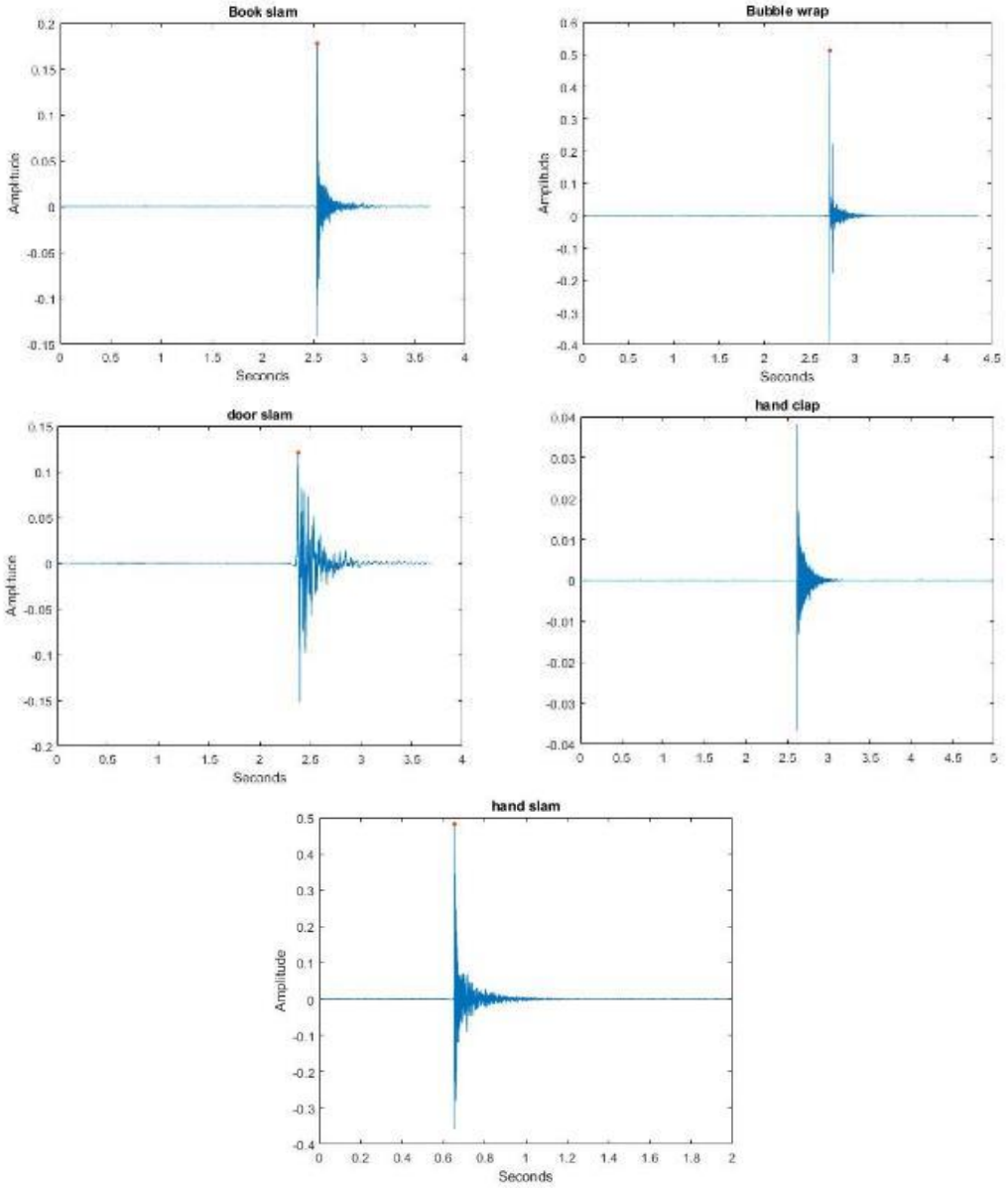

Fig. 23: Samples of False alarms

6.2. Computation of MFCC

We implemented the Mel frequency Cepstral Coefficient method to extract the features to be used for classification. The method involves the following:

- Define a moving frame which moves with a particular overlapping frame which would divide the data into smaller frames for easier processing. In our case, the frame length was for 1024 samples (for 50 ms), 512 samples (for 30 ms) and 256 samples (for 15 ms) along with an overlapping frame length of about 10 ms.
- Apply the hamming window on the different frames and perform the FFT on the convolution.
- Using the Fourier transform, get the power spectrum by squaring the absolute value of the transform.
- Derive the mel spectrum by convolution of the filter bank and the power spectrum
- Perform the logarithm of the mel spectrum
- Apply Discrete Cosine transformation on the logarithm of the mel spectrum
- Discard the higher order mel frequency coefficients.

The mel frequency filter bank was constructed in this case to have the same amplitude and have a geometrically progressive frequency distribution. We used the triangular filter bank because we are trying to approximate the non-linear response of the human auditory system and map the frequencies to the mel scale. It also helps because it is relatively easy to implement since we would need lesser number of coefficients and therefore less computation power.

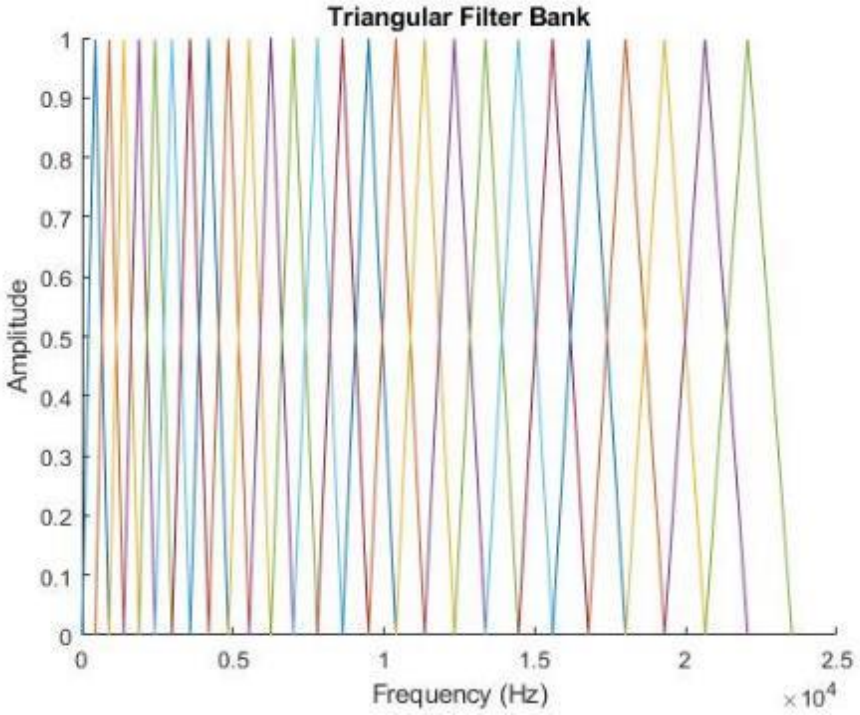

Fig. 24: MFCC Filter bank

6.3. Computation of IMFCC

The steps for running the computation of the MFCC except for the filter bank stage. The inverse mel frequency filter bank basically works by reversing the mel scale. This means we would be more interested in the higher frequencies and be extracting more features from that range. We would also be using triangular filter banks for comparable results as well as similar frequency ranges used for the feature extraction. The inverse mel filter bank tends to work better than mel filter bank by providing less distortion as well as is known for performing better in situations with low signal-to-noise ratio.

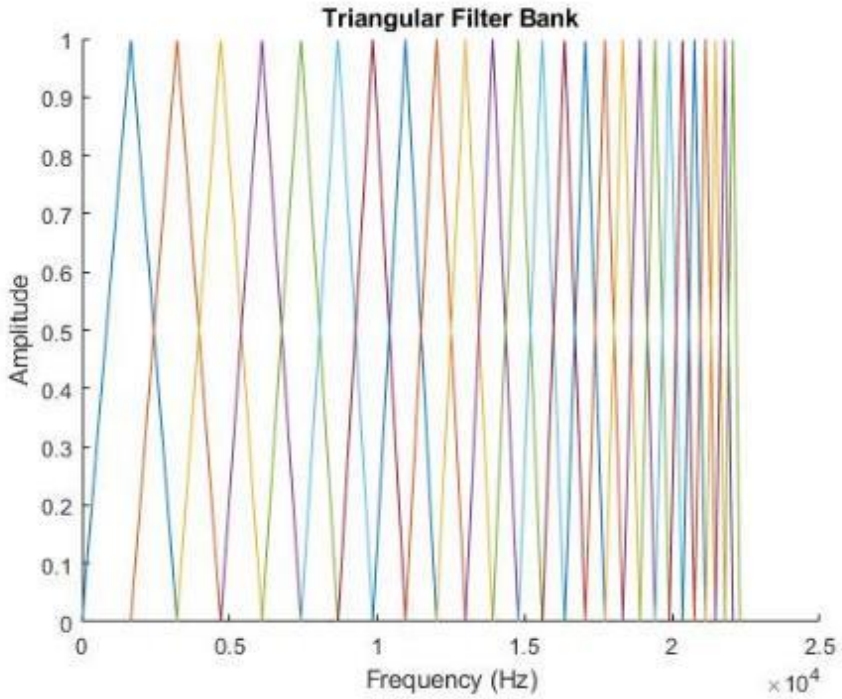

Fig. 25: IMFCC Filter bank

6.4. Computation of LFCC

The computation of LFCC is like both MFCC and IMFCC with the only difference being the linear scale of distribution of the frequencies. The linear filter bank has been proven to work better than the mel filter bank for feature extraction in some certain scenarios and tends to be more robust towards noise as well as other environmental factors.

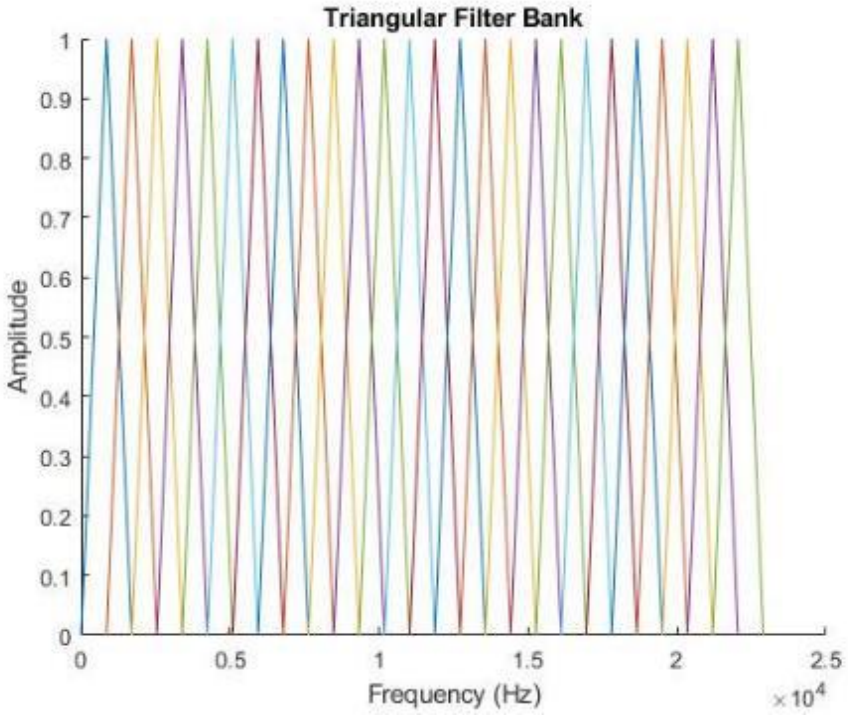

Fig 26: LFCC Filter bank

6.5. Computation of GTCC

We implemented the Gammatone frequency Cepstral Coefficient method to extract the features to be used for classification. The method involves the following:

- Define a moving frame which moves with a particular overlapping frame which would divide the data into smaller frames for easier processing. In our case, the frame length was for 1024 samples (for 50 ms), 512 samples(for 30 ms) and 256 samples (for 15 ms) and an overlapping frame length of about 10 ms.
- Apply the hamming window on the different frames and perform the FFT on the convolution.
- Using the Fourier transform, get the power spectrum by squaring the absolute value of the transform.
- Derive the gammatone spectrum by convolution of the filter bank and the power spectrum
- Perform the logarithm of the gammatone spectrum
- Apply Discrete Cosine transformation on the logarithm of the gammatone spectrum
- Discard the higher order gammatone frequency coefficients.

The gammatone filter bank consists of gaussian distributed filters which have a geometrically increasing variance about them which is pretty similar to the frequency distribution for its mel counterpart. Gammatone tends to give a more accurate representation of human hearing than the mel filter banks does due to the narrower bandwidth. The gammatone filters are also less sensitive to noise and other distortions.

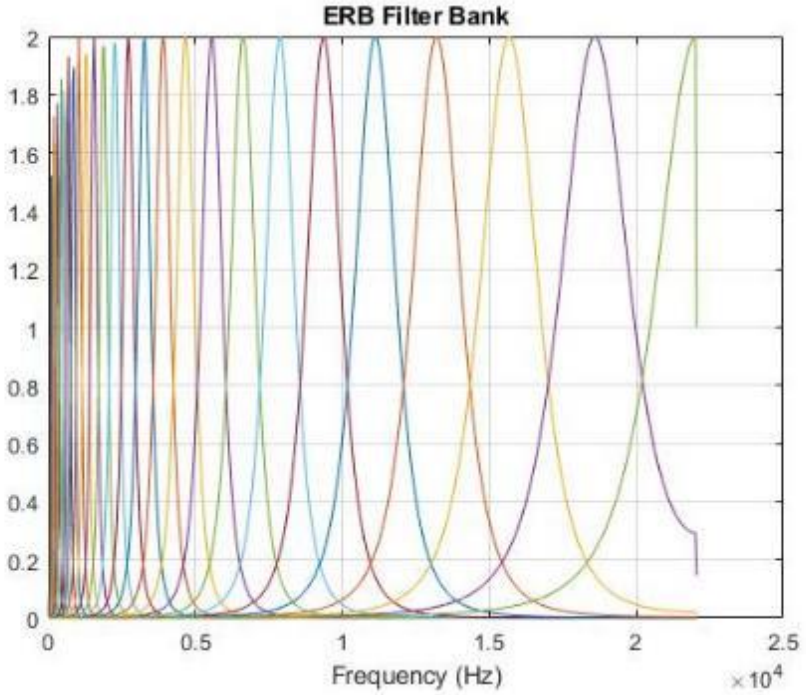

Fig. 27: GTCC Filter bank

6.6. COMPUTATION OF CLASSIFICATION

For the classification, we would take in the 26 features each extracted from MFCC, IMFCC, LFCC and GTCC which were all labelled either 0 for false alarms, 1 for 9 mm or 2 for 5.56 mm. Using these features and the label we were able to use a multi label non-linear classification algorithm. This algorithm was Support Vector Machines (SVM) with polynomial kernel of power 2. The classification was done across 3 different frame lengths (15 ms, 30 ms, 50 ms) with 4 different feature extraction methods (MFCC, IMFCC, LFCC, GTCC) to see which frame works best along with which method would produce the best results. A data structure having the gunshot frame, time of the peak, max value, mel coefficients, inverse mel coefficients, linear coefficients and gammatone coefficients was defined for easy implementation.

7. IMPLEMENTATION AND RESULTS

7.1. RESULTS USING SVM CLASSIFICATION

The data used for this project were audio recording of gunshots which were recorded in similar environments for 9 mm, 5.56 mm, Hand slams, book slams, hand claps, bubble wraps and door slams. The detection, feature extraction and classification algorithm were applied on the dataset which meant we had the detection and feature extraction for all the data but for the Support Vector Machines classification algorithm we would have to train the algorithm and then test with it.

A polynomial kernel of order 2 was used because it produced the best results experimentally compared to the linear kernel or even the gaussian kernel which were both tested. The training-testing data distribution was like 80% to 20% respectively across all classes and frame length. We had basically only three classes for this multi-label classification problem: 0 (false alarms), 1 (9 mm) and 2 (5.56 mm) across the different implemented frame lengths for detected gunshot signals of 15, 30 and 50 ms.

Table 1: The distribution of training samples for SVM classification

Classes	Frame length (ms)	Number of Samples
False alarms	15	86
	30	86
	50	86
9 mm	15	67
	30	70
	50	60
5.56 mm	15	45
	30	100
	50	100

We would then be testing the classification algorithm across the 3 different frame lengths and 4 classification methods. We would be testing using the SVM classification method with polynomial kernel using the following test data distribution:

Table 2: The distribution of testing samples for SVM classification

Classes	Frame length (ms)	Number of Samples
False alarms	15	21
	30	21
	50	21
9 mm	15	18
	30	31
	50	16
5.56 mm	15	16
	30	31
	50	31

For the proper interpretation of the results, there is use of some specific metrics to help better understand what is going on with the model. These metrics are:

$$Accuracy: ACC = \frac{TP + TN}{TP + FP + TN + FN} \tag{33}$$

$$Precision: PRC = \frac{TP}{TP + FP} \tag{34}$$

$$Recall: RLC = \frac{TP}{TP + FN} \tag{35}$$

$$MCC = \frac{Matthews\ Correlation\ Coefficient: \quad TP.TN - FP.FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{36}$$

$where\ T\ stands\ for\ True, F\ stands\ for\ False, P\ stands\ for\ Positive\ and\ N\ stands\ for\ Negative$

We ran the tests with the classification algorithm, and we have the following results in the form of confusion matrices and tables:

Table 3: Confusion Matrices for SVM classification of 15 ms gunshot data

MFCC				
	ACTUAL			
PREDICTED		**0**	**1**	**2**
	0	14	2	2
	1	5	18	4
	2	2	0	10

IMFCC				
	ACTUAL			
PREDICTED		**0**	**1**	**2**
	0	15	2	2
	1	4	18	4
	2	2	0	10

LFCC				
	ACTUAL			
PREDICTED		**0**	**1**	**2**
	0	14	2	2
	1	5	18	4
	2	2	0	10

GTCC				
	ACTUAL			
PREDICTED		**0**	**1**	**2**
	0	21	1	0
	1	0	18	4
	2	0	1	12

Table 4: Confusion Matrices for SVM classification of 30 ms gunshot data

MFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	20	3	0
	1	0	21	8
	2	1	7	23

IMFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	20	4	1
	1	0	21	7
	2	1	6	23

LFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	21	3	2
	1	0	22	6
	2	1	6	23

GTCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	21	1	0
	1	0	24	0
	2	0	1	31

Table 5: Confusion Matrices for SVM classification of 50 ms gunshot data

MFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	16	2	1
	1	4	12	2
	2	1	2	28

IMFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	17	2	1
	1	2	13	2
	2	2	1	28

LFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	18	2	1
	1	2	12	2
	2	1	2	28

GTCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	21	1	0
	1	0	15	0
	2	0	0	31

Based on the confusion matrices derived above, we can clearly see some patterns with the methods and using the proper metrics of accuracy, recall, precision, and Matthews' Correlation coefficient we are able to look much more closely to correctly understand the effectiveness of each feature extraction and frame length to define which is much more useful in our case as seen in the following tables:

Table 6: Table of Accuracy, Precision and Recall of each test implemented for SVM classification

Classes	Frame length (ms)	Feature Extraction	ACC (%)	PRC (%)	RLC (%)
False alarms	15	MFCC	80.70	66.67	77.78
		IMFCC	82.46	71.43	78.95
		LFCC	80.70	66.67	77.78
		GTCC	98.24	100	95.45
	30	MFCC	95.18	95.24	86.96
		IMFCC	92.77	95.24	80.00
		LFCC	92.77	95.24	80.00
		GTCC	98.71	100	100
	50	MFCC	88.06	76.19	84.21
		IMFCC	89.55	80.95	85.00
		LFCC	91.04	85.71	85.71
		GTCC	98.53	100	95.45
9mm	15	MFCC	80.70	90.00	66.67
		IMFCC	82.46	90.00	69.23
		LFCC	80.70	90.00	66.67
		GTCC	91.07	90.00	81.82
	30	MFCC	80.72	67.74	72.41
		IMFCC	79.52	67.74	75.00
		LFCC	81.93	70.97	78.57
		GTCC	97.44	96.00	100
	50	MFCC	85.29	75.00	66.67
		IMFCC	89.71	81.25	76.47
		LFCC	88.24	75.00	75.00
		GTCC	98.53	93.75	100
5.56mm	15	MFCC	85.96	62.50	83.33
		IMFCC	85.96	62.50	83.33
		LFCC	85.96	62.50	83.33
		GTCC	91.22	75.00	92.31
	30	MFCC	80.72	74.19	74.19
		IMFCC	81.93	74.19	76.67
		LFCC	81.93	74.19	76.67
		GTCC	98.72	100	98.96
	50	MFCC	91.18	90.32	90.32
		IMFCC	91.18	90.32	90.32
		LFCC	91.18	90.32	90.32
		GTCC	100	100	100

Table 7: Table of overall accuracy, precision, recall and Matthew's correlation coefficient for the test dataset

Frame length (ms)	Feature Extraction	Accuracy (%)	PRC (%)	RLC (%)	MCC (-)
15	MFCC	73.68	73.06	75.93	0.6110
	IMFCC	75.44	74.64	77.17	0.6352
	LFCC	73.68	73.06	75.93	0.6110
	GTCC	89.47	88.33	89.86	0.8429
30	MFCC	77.11	79.06	77.85	0.6537
	IMFCC	77.11	79.06	77.22	0.6565
	LFCC	78.31	80.13	78.41	0.6747
	GTCC	98.70	98.67	98.96	0.9805
50	MFCC	82.35	80.50	80.40	0.7268
	IMFCC	85.29	84.17	83.93	0.7714
	LFCC	85.29	83.68	83.68	0.7707
	GTCC	98.53	97.92	98.48	0.9774

7.2. RESULTS USING NEURAL NETWORK MODEL

The classification was also performed using Neural Networks with the intention of getting a possibly stronger classifier. The labels remained the same with 0 being false alarms consisting of hand slams, hand claps, door slams, bubble wrap as well as book slams, 0 for 9 mm gunshot signals and 1 for 5.56 mm gunshots. A similar test to that performed on the SVM classification model across different saved gunshot frames (15, 30 and 50 ms) and feature extraction methods (MFCC, IMFCC, LFCC, GTCC) for the neural networks.

A neural network of about 20 layers (an input layer, 18 hidden layers and an output layer) was trained with the following distribution of data to train the model for proper classification. It was a simple neural network with perceptron units. These layers were trained using the Levenberg-Marquardt training method and the neural network training tool in MATLAB.

Table 8: The distribution of training samples for NN classification

Classes	Frame length (ms)	Number of Samples
False alarms	15	86
	30	86
	50	86
9 mm	15	67
	30	67
	50	67
5.56 mm	15	45
	30	45
	50	45

There is a difference in the number of training samples of SVM and NN. This is because while training the SVM classifier, there were some classes that needed to be oversampled and others oversampled for proper classification. We use a comparable dataset for both training and testing for NN.

Now, we go through the process of optimizing and tunning the hyperparameters of the model such that we get very good training accuracies across different tests. The trained neural network is now tested with the 3 different frames across the 4 feature extraction algorithms implemented using the following test data distribution:

Table 9: The distribution of testing samples for NN classification

Classes	Frame length (ms)	Number of Samples
False alarms	15	21
	30	21
	50	21
9 mm	15	15
	30	15
	50	15
5.56 mm	15	16
	30	16
	50	16

Using MATLAB fitnet function for the different tests needed to be performed we can get the following results in the form of confusion matrices and the proper metrics as seen below:

Table 10: Confusion Matrices for NN classification of 15 ms gunshot data

MFCC				
ACTUAL				
PREDICTED		**0**	**1**	**2**
	0	16	5	6
	1	5	15	0
	2	0	0	10

IMFCC				
ACTUAL				
PREDICTED		**0**	**1**	**2**
	0	14	4	6
	1	7	16	0
	2	2	0	10

LFCC				
ACTUAL				
PREDICTED		**0**	**1**	**2**
	0	16	6	4
	1	4	14	2
	2	1	0	10

GTCC				
ACTUAL				
PREDICTED		**0**	**1**	**2**
	0	21	0	0
	1	0	19	0
	2	0	1	16

Table 11: Confusion Matrices for NN classification of 30 ms gunshot data

MFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	16	0	0
	1	5	19	1
	2	0	1	15

IMFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	18	2	0
	1	3	17	1
	2	0	1	20

LFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	15	3	2
	1	6	16	4
	2	0	6	10

GTCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	21	0	0
	1	0	19	0
	2	0	1	16

Table 12: Confusion Matrices for NN classification of 50 ms gunshot data

MFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	20	2	0
	1	1	18	0
	2	0	0	16

IMFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	20	3	0
	1	1	17	1
	2	0	1	15

LFCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	16	1	0
	1	5	18	6
	2	1	1	10

GTCC				
		ACTUAL		
		0	**1**	**2**
PREDICTED	**0**	21	0	0
	1	0	19	0
	2	0	1	16

From these confusion matrices, we can clearly see that Neural Networks tend to classify better than Support Vector Machines for some of these tests performed but also seems to classify better certain classes and has some interesting things to note as can be seen using the metrics defined as follows:

Table 13: Table of Accuracy, Precision and Recall of each test implemented for NN classification

Classes	Frame length (ms)	Feature Extraction	ACC (%)	PRC (%)	RLC (%)
False alarms	15	MFCC	71.93	59.26	76.19
		IMFCC	66.67	58.33	66.67
		LFCC	73.68	61.54	76.19
		GTCC	100	100	100
	30	MFCC	91.22	100	76.19
		IMFCC	91.22	90.00	85.71
		LFCC	80.70	75.00	71.43
		GTCC	100	100	100
	50	MFCC	94.74	90.91	95.24
		IMFCC	92.98	86.96	95.24
		LFCC	87.72	94.12	76.19
		GTCC	100	100	100
9mm	15	MFCC	82.46	75.00	75.00
		IMFCC	80.70	69.57	80.00
		LFCC	78.47	70.00	70.00
		GTCC	98.25	100	95.00
	30	MFCC	89.47	76.00	95.00
		IMFCC	87.72	80.95	85.00
		LFCC	66.67	61.54	80.00
		GTCC	98.25	100	95.00
	50	MFCC	94.74	94.74	90.00
		IMFCC	89.47	89.47	85.00
		LFCC	77.19	62.07	90.00
		GTCC	98.25	100	95.00
5.56mm	15	MFCC	89.47	100	62.50
		IMFCC	85.96	95.24	95.24
		LFCC	87.72	62.50	83.33
		GTCC	98.25	94.12	100
	30	MFCC	96.49	93.75	93.75
		IMFCC	96.47	95.24	95.24
		LFCC	82.46	90.91	62.50
		GTCC	98.25	94.12	100
	50	MFCC	100	100	100
		IMFCC	96.49	100	93.75
		LFCC	85.96	90.91	62.50
		GTCC	98.25	94.12	100

Table 14: Table of overall accuracy, precision, recall and Matthew's correlation coefficient for the test dataset for NN classification

Frame length (ms)	Feature Extraction	Accuracy (%)	PRC (%)	RLC (%)	MCC (-)
15	MFCC	71.93	78.09	71.23	0.5803
	IMFCC	70.18	75.97	69.72	0.5512
	LFCC	70.18	74.15	69.56	0.5511
	GTCC	98.25	98.04	98.33	0.9740
30	MFCC	87.72	89.92	88.31	0.8246
	IMFCC	88.71	88.73	88.65	0.8310
	LFCC	71.93	75.82	71.31	0.5810
	GTCC	98.25	98.04	98.33	0.9740
50	MFCC	97.40	95.22	95.08	0.9209
	IMFCC	91.23	92.14	91.33	0.8684
	LFCC	77.19	82.37	76.23	0.6728
	GTCC	98.25	98.04	98.33	0.9740

8. CONCLUSION

The thesis involved testing the detection and classification of gunshots across different parameters which were the frame lengths, the feature extraction method and the classification algorithm to see which is the most optimal case for each case. We would look at each of these parameters and decide which worked best across different cases interrelated with the other parameters.

8.1. FRAME LENGTH

The classification algorithm was tested across different frame lengths of 15, 30 and 50 ms. The results showed that the longest frame seemed to produce the best result with the 30 ms producing slightly worse results than the 50 ms frame and the 15 ms frame producing the worst of them all. This might be due to the fact that for proper classification, the feature extraction might need more signal information from the acoustic event to properly process the acoustic event. The smallest frame might actually be way too small because it basically only gives information about the muzzle blast and perhaps some immediate environmental reflections which is the main basis for our detection but we might need much more information from the event than just two parts such as the mechanical vibrations. From the results, we can see that this frame doesn't really classify either false alarms or gunshots effectively because with such small amount of information as can be seen from the diagrams they all seem pretty similar which might not help so much. The 30 ms frame works a bit better because now we have a bit more information and from the results, we can see that it can properly distinguish between a gunshot and a false alarm but once we get to specific gunshots, it doesn't seem to work well while the 50 ms frame produces the best results across feature extraction and classification algorithms.

8.2. FEATURE EXTRACTION

The thesis involved building a classification system across different acoustic feature extraction algorithms which were MFCC, IMFCC, LFCC, GTCC. The best algorithm amongst them was GTCC which produced very good classification results almost comparable across both frame lengths and machine learning classification algorithms. This is probably because the GTCC feature extraction algorithm tends to have a better auditory modelling than the other methods considered as well as its robustness to noise which we would obviously have in any recording environment and non-linear processing which helps to produce much better results even with limited amount of information provided by the detection frames we were using. MFCC and IMFCC were seen to produce pretty similar results with IMFCC producing just slightly better results then MFCC in most tests especially with SVM classification and it seemed to go the other way while performing the Neural Networks classification. This might be because for SVM which seems to need more information for proper classification, it might help to use also effectively filter some low frequency characteristics to get more information while for Neural Networks it tends to work better with limited information so it can work with majorly high frequency information such as the muzzle blast and the shockwave which might be enough to produce good enough results and the low frequency characteristics may not be so useful in this case. The LFCC method seemed to produce the worst result across all tests which might be due to its linearity. LFCC shouldn't be discounted because it seemed to produce comparable results with SVM but was really bad when tested with Neural Networks.

8.3. CLASSIFICATION

In this thesis, two machine learning classification algorithms were implemented which were Support Vector Machines (SVM) and Neural Networks (NN). The SVM classification algorithm was implemented with a polynomial kernel of order 2 and the NN classification algorithm was implemented with 20 layers (an input layer, 18 hidden layers and an output layers). The NN classification algorithm produced much better results across all frame lengths and all feature extraction algorithms except GTCC which had relatively similar results across almost all tests. The LFCC method seemed to produce the lowest set of results for the NN classification algorithm which is probably due to the linearity of the method which might not be useful in gunshot classification since we are looking for mostly high frequency characteristics. The GTCC method produced the best results with both MFCC and IMFCC producing similar results with MFCC slightly better. The 50 ms frame produces the best result since we have a lot of acoustic information across large frequencies to help get more distinguishable features than the 30 ms frame which had slightly better results than the 15 ms frame which was the worst. This means that for proper classification using NN, there was a need to properly detect the main muzzle blast or shock wave along with some more reflections and acoustic information about gunshot to which 50ms produces enough of but can still properly classify even with 15 ms frame or 30 ms frame. The SVM classification method seemed to not work so well with 15 ms and 30 ms frames across MFCC, IMFCC and LFCC but produced comparable results for all GTCC tests which is due to the properties of the feature extraction method. This is partly due to the methods themselves but also because of the kernel used which was a polynomial kernel of the order 2 which would be like that of a circular margin with a certain radius which is a good way to separate this non-linearly separable data but might not effectively capture the relationship between the features and the class especially in a small detection frame.

9. REFERENCES

1. J. Svatos, J. Holub, "Impulse Acoustics event detection, localisation and classification", IEE Transactions on Instrumentation and Measurement pp.99, doi: 10.1109/TIM2023.3252631
2. J. Svatos, J. Holub, "Smart Acoustic Sensor", in 5th International Forum on Research and Technologies for Society and Industry. Florence, IEEE, 2019. pp. 161-165, doi: 10.1109/RTSI.2019.8895591
3. R. C. Maher, "Modeling and signal processing of acoustic gunshot recordings," 2006 IEEE 12th Digital Signal Processing Workshop & 4th IEEE Signal Processing Education Workshop, Teton National Park, WY, pp. 257-261, 24-27 Sept. 2006.
4. R. C. Maher, "Acoustical characterization of gunshots," 2007 IEEE Workshop on Signal Processing Applications for Public Security and Forensics, "Washington, DC, USA, pp. 1-5, 2007.
5. Skye Gold, “These amazing GIFs show everything that happens when you fire a handgun,” Feb 17, 2015.
6. J. Svatos, J. Holub, J. Belak, "System for an acoustic detection, localisation and classification", Acta IMEKO, Vol. 10, No. 2, 2021, doi: 10.21014/acta_imeko.v10i2.1041.
7. Alfonso Chacón-Rodríguez, Pedro Julián, Liliana Castro, Pablo Alvarado, and Néstor Hernández, “Evaluation of Gunshot Detection Algorithms”, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS, VOL. 58, NO. 2, FEBRUARY 2011.
8. Pratheeksha Nair, July 24, 2018, *Medium.com,* accessed 12 December 2022, “The dummy’s guide to MFCC”, Meduim.com, <https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd>
9. *Practical Cryptography,* accessed 12 December 2022, <http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/ >
10. Howard Lei, Eduardo López Gonzalo ,“Mel, linear, and antimel frequency cepstral coefficients in broad phonetic regions for telephone speaker recognition”, Interspeech 2009
11. Ning Ma, *An Efficient Implementation of Gammatone Filters,* accessed 4 February 2023, <https://staffwww.dcs.shef.ac.uk/people/N.Ma/resources/gammatone/ >
12. *Linear Classification,* GitHub, accessed 4 February 2023, <https://leonardoaraujosantos.gitbook.io/artificial-inteligence/machine_learning/supervised_learning/linear_classification>
13. *Linear Classification,* accessed 15 February 2023, <https://mylearningsinaiml.wordpress.com/linear-classification/ >
14. Klaus-Robert Muller, ResearchGate, accessed 17 February 2023, <https://www.researchgate.net/figure/Linear-classifier-and-margins-A-linear-classifier-is-defined-by-a-hyperplanes-normal_fig4_221095188 >
15. *What are neural networks,* IBM, accessed 23 March 2023, <https://www.ibm.com/topics/neural-networks#:~:text=Neural%20networks%2C%20also%20known%20as,neurons%20signal%20to%20one%20another. >
16. *What Is A Neural Network,* AWS, accessed 11 April 2023, <https://aws.amazon.com/what-is/neural-network/>
17. V. Yaremenko, M. R. Azimi-Sadjadi and J. Zacher, "Unattended Acoustic Sensor Systems for Source Detection, Classification, and Tracking," in IEEE Transactions on Instrumentation and Measurement, vol. 68, no. 2, pp. 344-354, Feb. 2019, doi: 10.1109/TIM.2018.2849458.
18. S Akhtar, M Elshafei-Abmed, and M.S. Ahmed, “Detection of helicopters using neural nets, ”IEEE Trans. Instrum. Meas. ,vol. 50, no. 3,pp. 749–756, Jun. 2001.
19. N. Wachowski and M. Azimi-Sadjadi, “Detection and classification of nonstationary transient signals using sparse approximations and Bayesian networks,” IEEE / ACM Trans. Audio, Speech, Language Process., vol. 22, no.12, pp. 1750–1764, Dec. 2014

20. A. Morehead, L. Ogden, G. Magee, R. Hosler, B. White and G. Mohler, "Low Cost Gunshot Detection using Deep Learning on the Raspberry Pi," 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 3038-3044, doi: 10.1109/BigData47090.2019.9006456.
21. T. Mäkinen, P. Pertilä, "Shooter localization and bullet trajectory, caliber, and speed estimation based on detected firing sounds," Applied Acoustics, volume 71, n. 10, 2010, p. 902–913.
22. B. Kaushik, D. Nance, K. K. Ahuja "A Review of the Role of Acoustic Sensors in the Modern Battlefield", 11th AIAA/CEAS Aeroacoustics Conference, 2005.
23. J. Millet and B. Baligand, "Latest Achievements in Gunfire Detection Systems", Battlefield Acoustic Sensing for ISR Applications (pp. 26-1–26-14) NATO RTO-MP-SET-107, 2006.
24. *Patented Gunshot Detector,* Amberbox, accessed 22 April 2023 <https://amberbox.com/detection/patented-gunshot-detection# >
25. *The Perceptron,* Cornell University, accessed 22 April 2023, <https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote03.html>
26. *What is a Neural Network,* Tibco, accessed 5 May 2023, <https://www.tibco.com/reference-center/what-is-a-neural-network>