ISTANBUL TECHNICAL UNIVERSITY ★ INFORMATICS INSTITUTE

CONTEXT AWARE AUDIO-VISUAL ENVIRONMENT AWARENESS USING CONVOLUTIONAL NEURAL NETWORK

M.Sc. THESIS

Giray YILLIKÇI

Department of Communication Systems

Satellite Communication and Remote Sensing Programme

Thesis Advisor: Prof. Dr. İbrahim AKDUMAN
Anabilim Dalı : Herhangi Mühendislik, Bilim
Programı : Herhangi Program

MAY 2019

ISTANBUL TECHNICAL UNIVERSITY ★ INFORMATICS INSTITUTE

CONTEXT AWARE AUDIO-VISUAL ENVIRONMENT AWARENESS USING CONVOLUTIONAL NEURAL NETWORK

M.Sc. THESIS

Giray YILLIKÇI
(705111019)

Department of Communication Systems

Satellite Communication and Remote Sensing Programme

Thesis Advisor: Prof. Dr. İbrahim AKDUMAN
Anabilim Dalı : Herhangi Mühendislik, Bilim
Programı : Herhangi Program

MAY 2019

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

KONVOLÜSYONEL SİNİR AĞI KULLARAK SES VE GÖRÜNTÜ ARACILIĞIYLA ORTAM FARKINDALIĞI

YÜKSEK LİSANS TEZİ

Giray YILLIKÇI
(70511019)

İletişim Sistemleri Anabilim Dalı

Uydu Haberleşmesi ve Uzaktan Algılama Programı

Tez Danışmanı: Prof. Dr. İbrahim AKDUMAN
Anabilim Dalı : Herhangi Mühendislik, Bilim
Programı : Herhangi Program

MAYIS 2019

Giray YILLIKÇI, a **M.Sc**. student of ITU Informatics Institute student ID **705111019**, successfully defended the thesis/dissertation entitled "**CONTEXT AWARE AUDIO-VISUAL ENVIRONMENT AWARENESS USING CONVOLUTIONAL NEURAL NETWORK**", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Dr. İbrahim AKDUMAN** ....
Istanbul Technical University

Jury Members : **Assoc. Prof. Mehmet ÇAYÖREN** ....
Istanbul Technical University

Assist. Prof. Dr. Evrim TETİK ....
Istanbul Aydin University

Date of Submission : 3 May 2019
Date of Defense : 11 June 2019

To Science and Yoga

FOREWORD

First of all, I am thankful to my graduation project guide Professor Doctor Ibrahim AKDUMAN and Associate Professor Mehmet CAYOREN who is answering all my questions for encouraging me to undertake this project with their support and for providing necessary guidance concerning projects implementation. Without their superior knowledge and experience, the project would like in quality of outcomes, and thus their support has been essential.

May 2019

Giray YILLIKÇI
(Physics)

TABLE OF CONTENTS

Page

ABBREVIATIONS

MFCC	**:** Mel Frequency Cepstral Coefficient
TDOA	**:** Time Difference Of Arrival
GCC	**:** Genaralized Cross Correlation
PHAT	**:** Phase Transform
CNN	**:** Easily Convolutional Neural Network
AEC	**:** Acoustic Echo Cancellation
STFT	**:** Short Term Fourier Transform
SNR	**:** Signal to Noise Ratio
DCT	**:** Discrete Cosine Transform

LIST OF FIGURES

Page

AWARE AUDIO-VISUAL ENVIRONMENT AWARENESS USING CONVOLUTIONAL NEURAL NETWORK

SUMMARY

In this thesis, in most intelligent machine perception, sensing approaches require a novel approach to diminish computational burden over the system to increase the outcome object detection, object tracking, developed to understand the environment.

The main area of the study is to give real-time audio classification whose outputs would feed to the visual classification algorithms designed in the form of active audio-visual perception. The detection, localization, and tracking of the sound source are the main goal of audition and vision perception independently. Multiple signal classification based on Convolutional Neural Network method has employed for sound source localization and classification in audition modality.

An audio-visual pipeline has introduced for enhancing the awareness of visual classification and tracking by introducing auditory classification and direction estimation for the sound emitters in the surrounding environment, for this purpose, trending visual machine learning algorithms that have very high successive rates selected as the one end of the pipeline. At the starting of the pipeline, the surrounding sounds have classified, and if any of them are subject to track, the camera mounted servo rotated to the estimated direction for looking any object that would be the source emitter. The pipeline uses the very same CNN for also visual object detection which optimizes the computational load on the overall system.

KONVOLÜSYONEL SİNİR AĞI KULLANARAK SES VE GÖRÜNTÜ ARACILIĞI İLE ORTAM FARKINDALIĞI

ÖZET

Bu tezde tıpkı yaşayan canlılarda olduğu gibi kulakların ve gözünlerin beraber çalışması elektronik ortamda gerçekleştirilmesi yapılmıştır. Çoğu gelişmiş canlıda olduğu gibi gözlerin görüş açısı 160 dereceyi geçmemektedir. 360 derece görme yetisinin olmaması beyinde görüntü işlemenin çok yük getirmesindendir. Gözlerin görmediği açılarda duyu organlarından yararlanarak ortam farkındalığı elde edilir. Tıpkı duyulan sese doğru görüş açımızın çevrilmesi gibi bu çalışmada canlılardaki duyuların beraber çalışması yaklaşımı araştırılmıştır.

Duyulardan gelen bilgileri entegre ederek ortam farkındalığı edinmenin başlıca üç avantajı vardır. Öncelikle iki basit farklı ortam sensörünün beraber çalışarak yaptığı anlamlandırma yüksek başarımlı bir sensörün yapacağı anlamlandırmadan özellikle bilinmeyen ortamlarda daha yüksektir. Sistem bu şekilde daha esnekleşerek bilgi edinebilceği ortam yelpazesini genişletmektedir.

İkinci olarak iki göreceli daha basit sensörün işlem yükü yüksek başarımlı bir sensörden daha az yük getirmektedir. Böylece dış ortamlarda daha uzun çalışabilir ve maliyeti daha düşük olmaktadır.

Son olarak entegre çalışan sensörlerden biri devre dışı kalsa dahi ortamdan kısıtlı olsa da farkındalık alınabilmektedir ancak tek sensörlü sistemlerde olacak bir bozukluk tüm sistemin devre dışı kalmasına sebebiyet verecektir.

Özellikle son zamanlarda teknolojiye artan ilgi ve kendin yap akımı ile bir çok işlemci ve benzeri elektronik ürünler kolaylıkla bulunabilir hale gelmiştir. Buna ek olarak gerçek zamanlı görüntü ve ses işleme algoritmaları son yıllarda büyük yol almış olması ve tasarımdan prototiplemeye geçişte zaman-maliyet ikilisini aşağa çekilmesi tasarımları pratikte gerçeklemeye imkan vermiştir.

Çalışmada yüksek başarımlı görüntü işleyen yapay sinir ağları algoritmalarına işitsel bilgi öncülük ettirilerek farkındalık arttırımı sağlayan bir akış anlatılmaktadır. Çoklu mikrofon seti ile yön ve içerik tanıma yapılmaktadır.

Mantık akışında öncelikle çevre görültüsü dinlenerek sınıflandırılacak sesler arasında takip edilmesi gereken bir ses duyuldugunda, sesin geliş açısı çoklu mikrofon seti ile belirlenir. Sonrasında aranan sesin kaynağını görsel olarak bulabilmek için servoya bağlı kamera hesaplanan yöne doğru döndürülür.

Ses kaynağına aday görsel objeler bu mekanizma ile diğer aday objelerden ayıklanarak daha yüksek başarımlı tanıma ve takip yapabilme olanağı sağlar. Ses ve görüntü sınıflandırma için aynı Konvolüsyon Sinir Ağı kullanılarak sisteme yüklenen hesaplama yükü en elverişli halde kullanılır. Bu amaç için ses bilgisi spektrogramlara çevirilerek sesin resimselleştirilmesi sağlanmıştır. Tüm sınıflandırma ve obje tespitleri gerçek zamanlı yapılmaktadır.

Gelen ses yönünün tahmini yapmak için bir düzlemde bir karenin dört köşesine yerleştirilmiş 4 mikrofondan yararlanılır. Gelen sesin yönünü belirlemesi mikrofonlara aynı ses sinyalinin varış zamanlarının farkının hesaplanmaktadır. Yalnız belirtilmelidir ki bu yaklaşım sadece impals sesler için yeterliyken süreklilik gösteren seslerin aynı başarımla yön tahmini yapabilmek için mikrofonlara varan seslerin çapraz korelasyonlarının hesaplanması gerekmektedir. Böylelikle sesin geldiği yön bu sayede doğrulukla hesaplanır. Sesin kaynağından çıkış zamanı bilinmediğinden koordinatı hesapanamaz ancak geldiği açı hesaplanabilmektedir.

Buna ek olarak ses dalgalarını geliş açısının tespitini olumsuz etkileyecek olan yankı ve gürültü giderme algoritmaları prototiplemede kullanılan Respeaker V2 kartında bulunmaktadır. Bu sayede alt paragrafta anlatılan ses sınıflandırılması için temiz veri sağlamaktadır.

Ses tanıma sistemi algoritması için öncelikle seslerin zaman boyutunda kayıt edilmiş ses örneklerinin yapay zeka eğitimi için resimleştirilmesi gerekmektedir. Resimleştirmenin başlıca nedeni kullanılan konvolüsyonel sinir ağının iki boyutlu tercihen renkli girdilerle çalışmasıdır. Ses örnekleri olarak UrbanSoun8K veriseti kullanılmış olup 10 farklı sınıftan oluşan örnekler içerir. Bu 10 sınıf şöyledir. Köpek havlaması, siren, çalışan motor, oynayan çocuklar, klima, silah, çekiç, korna, matkap ve sokak müziği seslerinden oluşmaktadır.

Resimleştirme işlemleri için belli adımların izlenmesi gerekmektedir. Zaman boyutunda kayıt edilmiş sesler 4 saniye zarfına sığacak şekilde ayarlanır. Ayrıca seslerin hepsi 16 bitlik wav formatı olacak şekilde saklanmıştır. Ayarlanan ses örnekleri insan kulağının daha hassas olduğu Mel frekanslarındaki frekans aralıklarının ses örneklerinde yükseltilmesi ve daha az belirginlikteki frekansların ise sönümlenmesini yapar. Sonrasında Mel frekanslarının güçlendirildiği ses örnekleri 50 milisaniyelik bölümler halinde kısa süreli Fourier dönüşümünden geçirilir.

Fourier dönüşümünden geçirilen bölümler arka arkaya eklenerek ses örneğinin spektrogramı çıkartılır. Spektrogramların şiddet aralıklarını normalize etmek için doğrusal şiddet gösteriminden logaritmik şiddet gösterimine çevrilir. Logaritmik çevrimi yapılan sesler resimsel tanımada kullanılan konvolüsyonel sinir ağı eğitimi için kullanılır.

Çalışmada obje tespiti kısmı ses sınıflandırılmasında olduğu gibi konvolüsyonel sinir ağı kullanılarak yapılmıştır. Çalışma kapsamında tek görüntü karesinden çoklu kutu tespitinden yararlanılmıştır. Bu yöntemle görüntü değişik boyutlardaki özellik çıkarıcılarla etkileştirilerek büyükten küçüğe ve uzundan yayvana matriksler oluşturulur. Oluşturulan matrikslerin içinde obje adaylarının puanlamaları çıkarılır ve en yüksek puana ait obje bahsi geçen pencere içinde bulunmuştur çıkarımı yapılır.

Obje tespitinde ek olarak görüntüye bir sınıf atamaktansa görüntü üzerinde çeşitli boyutlarda kullanılan pencere methodu ile bulunan obje adayları birden fazla farklı objenin aynı görüntü üzerinde bulunması sağlar. Ancak aynı obje için birden fazla bulunan obje adaylarının ayıklanması gerekmektedir. Bu amaçla yakın komşuluklardaki ve aynı objeye ait obje adaylarının en yüksek yüzdelisi dışındaki diğer obje adayları görmezden gelinir.

Yapılan sadeleştirme işlemi ile aynı objenin birden fazla işaretlenmesinin önüne geçilir. Kameradan alınan video akışında tespit edilen objeler saklanır ve ses eşleştirilmesi için program akışına dahil edilir.

Çalışma akışındaki son aşama sınıflandırılan ses ile bulunan objenin birbirlerine etiketlenmisidir. Ses sınıfları ve obje tipleri arasında kurulan ilişki matriksine bakılarak önceklikle ses sınıfına ait obje tipi belirlenir sonrasında kameranın döndüğü yönde bahsi geçen objenin olup olmadığı kontrol edilir, eğer obje bulunursa bulunan obje ilişki matriksiyle eşlendirilir. Sonuç olarak eşleme ses sınıfı ve obje tipi olarak ekrana bastırılır. Eşleştirilme başarısız olsa dahi sınıflandırılan ses ve bulunan obje yazdırılır ancak eşleme oluşmadığı yazdırılır.

1. INTRODUCTION

1.1 Purpose of Thesis

In the last decade, new technological advancements have achieved in a way that no other era would have seen: the data science and data analytics, artificial intelligence now entering every aspect of the technological application. Nowadays people can take it as to standard and commonly used technologies such as machine learning, deep learning, and artificial intelligence applications. Additionally, these newly emerging technologies are getting cheaper every year thanks to consumer interest among these files and demand to the vast chip designers to catch up smartphone ecosystem. Most of the approaches are mimic the human or animal nature to come up with solutions that are seeking to find answers. Bio-inspired solutions have seen in most of the neural networks that in use for machine learning applications. This Thesis originated from the idea that combining human hearing, getting the direction of the exciting sound and if necessary seeking the sound source visually by a look at the course of the sound source and extracting valuable information from audio recordings.

Convolutional neural networks are commonly used systems for image classification problems. By exploiting properties of the input layer of a convolutional and adjusting the sound signal properties as an image it has shown that Convolutional Neural Networks are a good candidate for audio identification problem domains. This master thesis directs to contribute to creating a combination of content-based object tracking inspired by humans and animals. The pipeline shown in the Figure 1.1 gives the mechanism for only activates the visual search and tracking if and designated objects sound has heard.

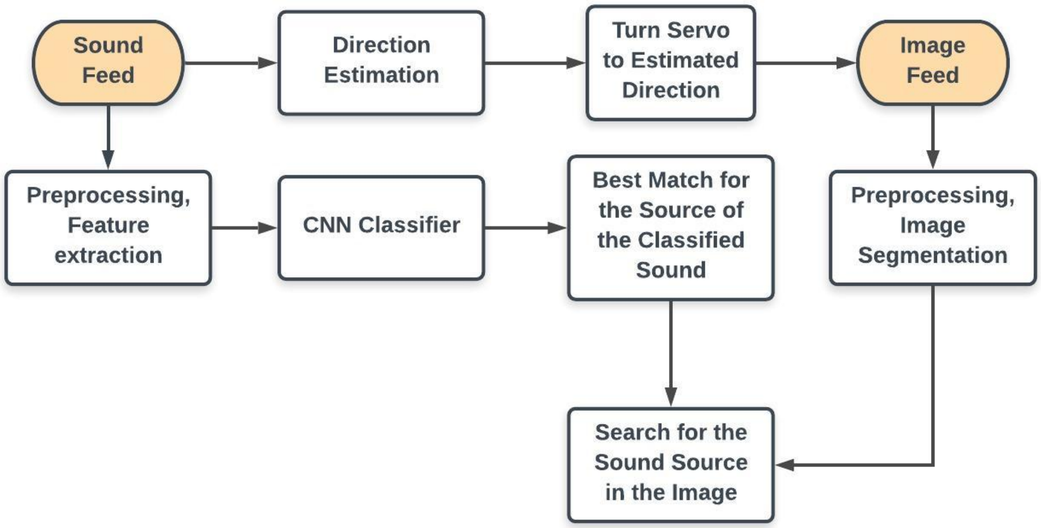

Sound Feed
Direction Estimation
Turn Servo to Estimated Direction
Image Feed
Preprocessing, Feature extraction
CNN Classifier
Best Match for the Source of the Classified Sound
Preprocessing, Image Segmentation
Search for the Sound Source in the Image

Figure 1.1 : Diagram of the working Mechanism of Sound and Image Classifiacation Integration

1.2 Literature Review

The topic of environmental sound classification using artificial neural networks has drawn increasing interest over the last few years. In[1] and[2], the authors used convolutional neural networks for environmental sound classification, making use of the public Urbandsound8K dataset, which is also the dataset used for this study. Using different architectures, they achieved average test accuracy results of 73% and 79% individually, the latter using different data augmentation methods. The authors of used two popular convolutional deep neural networks for image recognition, AlexNet, and GoogleNet, using the spectrogram of the audio signals as input feature to the networks. In[3] the authors used deep recurrent neural networks for the same purposes of audio scene classification achieving state-of-the-art accuracy results in the LITIS dataset, reinforcing the legality of this type of architecture for the audio classification task. In[4] the authors approached the same enigma from a multi-task point of view, suggesting that recognition accuracy may improve when using a multi-task model rather than multiple task-specific models. Although it can be seen that extensive research has been carried out on only audio classification, very small attention has been paid to the task of audio-visual classification, where the goal is to predict not only by hearing but also by seeing, just like sophisticated living beings do. Despite signifying an apparent real-world problem audio and visual classification havent use the same Convolutional Neural Network in real-world applications.

2. MULTILATERATION

Multilateration is an estimation procedure given the calculation of the difference in separation to two stations at known areas by transmitting signals at known conditions. Not at all like calculations of complete separation or point, estimating the difference in remove within two stations brings about a vast number of areas that fulfill the estimation. At the point when these conceivable fields are outlined, they shape a hyperbolic curve. To find the right area along that curve, multilateration depends on various calculations: a moment estimation taken to another combine of stations will create a moment bend, which converges with the first. At the point when the two curves are investigated, few convincing areas are revealed. The illustration given in Figure 2.1.

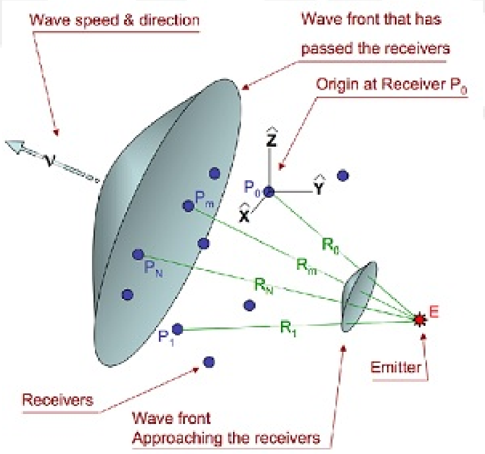


Figure 2.1 : Multilateration with single emitter multiple receivers[14]

Multilateration is a common approach in signaling route structures, where it is known as the hyperbolic route. These structures are usually simple to elaborate as there is no demand for a typical time-marker, and the difference in the timing can be calculated unmistakably using an oscilloscope. This framed the proof of various usually employed route frameworks beginning in World War II with the British Gee structure and a few similar structures presented during the following few decades. The presentation of the chip significantly streamlined task, enormously growing fame among the 1980s. Different structures keep on being employed, yet the unlimited utilization of satellite track structures like GPS have made these structures to a large degree.

It is extensively used in civil and military purposes to precise position a vehicle or aircraft, stationary emitter by estimating the "time difference of arrival" (TDOA) of from the emitter at three or more synchronized receiver ends, surveillance application, or the signals from three or more synchronized emitters at one collector area route application.

2.1 Finding A Transmitter from Multiple Recepient Sinks

The primary one is the reconnaissance application due to the finding a transmitter from different collector locales. If a beat has emitted from a stage, for the most part, arrive at unique times at two spatially isolated recipient destinations, the TDOA being due to the distinct separations of each recipient from the stage. In reality, for given areas of the two collectors, an entire set of emitter areas would grant the same estimation of TDOA. Given two recipient areas and a known TDOA, the locus of conceivable emitter areas is one-half of a two-sheeted hyperboloid.

If necessary, with two collectors at known areas, an emitter can be found onto a hyperboloid. Note that the recipients do not ought to know the outright time at which the beat has not transmitted – as it were the time distinction is required.

The 3rd receiver at a 3rd location. This could give one more axis TDOA measurement. There is a 3rd TDOA; however, the first one is dependent on the first two TDOA and does not give further dimensionality. So, the source is located on the curve calculated by the two intersecting hyperboloids. A fourth receiver has needed

for another orthogonal TDOA, and this will give an additional hyperboloid, the crossover of the spline with this hyperboloid provides up to two more solutions to the problem. The source is then positioned at least one or at one of the solutions.

With four sensors, there square measure three independent TDOA, three autonomous parameters square measure required for a degree in the 3-dimensional area. Also, for many groups, three independent TDOA can still provide 2 points in 3D space. With extra receivers, increased accuracy has often obtained. For instance, GPS the atmosphere will influence the traveling time of the signal, and additional satellites will provide much correct location. For associate overdetermined constellation, a statistical procedure is often used for 'reducing' the errors. Extended Kalman filters square measure used for rising the proper signal arrival times. Averaging over extended times may also improve accuracy. The accuracy additionally improves if the receiver’s square measure placed in an exceeding. The emitting platform could, or may not, get together within the multilateration surveillance progressions.

2.2 Acoustic Echo Cancelation

Echo cancellation methods are regularly called acoustic echo cancellation, AEC. Most of the time, as there are various types and causes of echo with individual components, including acoustic echo. For example, sounds from a speaker recorded by a receiver, which can change considerably over time and which changes much smaller than an acoustic echo. In practice, the same methods are used to handle all types of sound echo so that an acoustic echo canceller can shrink acoustic echo. AEC, in precise, is usually employed to cover to echo cancelers in common, despite whether they were intended for the acoustic echo.

Although echo cancellers have similar aim preventing a speaking person from hearing an echo of their voice the methods, they use are different:

Echo cancellation is the first identifying the originally broadcasted signal that re-appears, with some lag, in the transmitted or collected signal. Once the sound echo is known, it can be eliminated by deducting it from the exchanged an audio signal. This method is usually executed digitally utilizing a digital signal processor, or it can be implemented in analog circuits as well.

The acoustic echo cancellation process has multiple steps are described below:

A last-received signal delivered to the system.

The last-received signal reproduced.

The last-received signal has filtered and delayed resembling the first-received signal.

The filtered last-received signal has subtracted from the first-received audio signal.

The resultant signal describes sound being in the room excluding either straight or reflected audio sound.

The principal challenge for an echo canceller is identifying the nature of the filtering to do applied to the last-received signal such that it be similar to the concluding first-received signal. The filter is necessarily a model of speaker, microphone and the room's acoustical qualities. Echo cancellers needs to be dynamically adjustable because the features of the speaker of first-received and microphone are generally not recognized in advance. The acoustical characteristics of the room of near end are also not usually known in advance. If the microphone moves moved relative to the speaker the room causing variations in the acoustic reflections. By using the far-end signal as the perturbation, modern systems adaptively filter for converging from zero to 60 dB of cancellation in around less than a fraction of a second.

The current sound echo cancellation generally applied to the human sound bandwidth of phone circuits. Transmit frequencies within 0,3K Hz and 3K Hz, the range requisite for speech coherence. Video conferencing is one domain where full bandwidth audio is exchanged. In this case, specialized solutions are engaged to perform echo cancellation.

Echo suppression may have the adverse outcomes of removing true signals from the transmission. This can create a signal loss called 'clipping'. However, the effect is more like a 'squelch' than magnitude clipping. In an ideal state, then, sound source echo cancellation only will be used. However, this is unsatisfactory in many applications, distinctly software phones on networks with a large impediment and narrow throughput. Echo cancellation and suppression would perform in association to achieve acceptable performance. Acoustic echo canceller shown in the Figure 2.2.

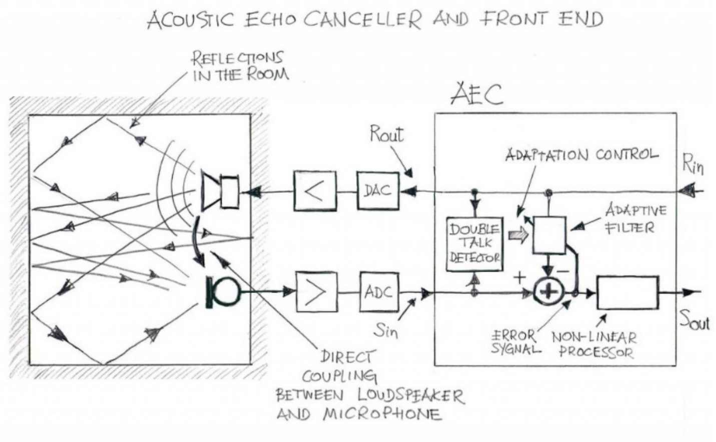


Figure 2.2 : Acoustic Echo Cancellation[13]

2.3 Frequency Generalized Cross Correlation Phase Transform

The computation of the TDOA within all of the channels and the associated carrier has repeated along the recording in order for the beam-forming to respond to changes in the emitter. In this application, it has computed every one-fourth of a second. This phenomenon called segment size or analysis scroll over a window of half of a second called the analysis window which covers the current analysis segment and the next. The breadth of the filter window and the section size constitute a tradeoff. A broad analysis window or segment window lead to a reduction in the analysis of changes in the TDOA. On the contrary, using a small examination window decreases the robustness of the cross-correlation calculation. Lesser acoustic frames are employed to compute it[8]. The reduction of the segment size also grows the computational burden of the system, while not growing the quality of the output audio signal.

In order to compute the TDOA within the associating channel section, a typical way to calculate it as the postponement that reasons the cross-correlation within the two signals segments to be maximum. In order to improve robustness against reverberation, it is standard practice to use the Generalized Cross Correlation with Phase Transform as presented by Velasco et al[5].

Given signals $Z_i(f)$ and $Z_j(f)$ the GCC-PHAT is defined as:

$$G_{PHAT(f)} = \frac{Z_i(f)[Z_j(f)]^*}{|Z_i(f)[Z_j(f)]^*|} \tag{2.1}$$

Where signals $Z_i(f)$ and $Z_j(f)$ are the Fourier transforms for the two signals and $\left[Z_{i,j}(f)\right]^*$ implies the complex conjugate. The TDOA for these two microphones has calculated as:

$$d_{PHAT(i,j)} = argmax\ (R_{PHAT}) \tag{2.2}$$

(R_{PHAT}) is the inverse Fourier transform of Eq.2.2.

Although the highest value of matches to the calculated TDOA for that particular section, there are three distinct cases for that it was considered not right to use the absolute maximum from the function. On the one hand, the peak can be due to a artificial noise or effect not related to the speaker active at that time in the neighboring auditory region, being the speaker of embodied by some other local maximum of the cross-correlation.

On the contrary, when multiple speakers are overlaying one another, every speaker will be represented by a peak of the cross-correlation. However, the absolute peak may not be continuously assigned to the same speaker, resulting in artificial speaker toggle. In order to efficiently magnify the signal, it would be maximum first to detect when more than single speaker is delivering at the same time and then obtain a filter and then sum signal for each one, supporting the picked lags and withdrawing them from speaker toggle. Due to a lack of an adequate overlap detector, this was not applied in this thesis and continued as future work.

Also, when the section that has processed has saturated with either noise or irregular acoustic events data the GCC-PHAT function obtained will not be at all informative. In such a case, no emitter delay data can be extracted from the signal and the delays should be eliminatd and replaced by something more informative.

In the system implementation, to deal with such issues, the top relative peaks in eq. 2.2 are computed, and several delay post-processing techniques are applied to maintain and choose the suitable delay before syncing the audio signals.

2.4 Three-Dimensional Direction Estimation

For finding the designated area in two-dimensional geometry, one can, for the most part, adjust the utilize for the 3-D geometry. Furthermore, there are specific calculations for two-measurements prominent are the techniques distributed by Fang for a Cartesian plane.

The states provide a calculation for each sound source direction [6]. This estimated direction $\emptyset_i^l$corresponds to the first moment of the latter random variable $(\acute{d})_i^{l|l}$:

$$\emptyset_i^l = \iiint N((\acute{d})_i^{l|l}\, \boldsymbol{H}\acute{\boldsymbol{x}}_i^{l|l}, \boldsymbol{H}\boldsymbol{P}_i^{l|l}\boldsymbol{H}^T)(\acute{d})_i^{l|l} dxdydz \qquad (2.3)$$

Then simplyfies to

$$\emptyset_i^l = \boldsymbol{H}\acute{\boldsymbol{x}}_i^{l|l} \qquad (2.4)$$

3. SOUND CLASSIFICATION

Sound processing is another aree where this study focuses after estimation for a sound source direction. For processing sound, based information that is a sequence, one needs to use useful methods to extract information. For this purpose, Mel-Frequency Cepstral Coefficient [9] is a common way to features nowadays. In this chapter, MFCC [7] and filter banks are used and will be introduced in a detailed way for employing the extracted features for the calculations.

Calculation of MFCC and filter banks have some similarities a sound signal feeds in to a pre-emphasis filter which dims the low-frequency component of a signal and lighten up the high-frequency signals then the signal is divided in to very short intervals, and a particular function has applied in the purpose of making the frame ready for Short-time Fourier Transform. After the application of the STFT power spectrum is taken as an output. In the nex precess filter banks have computed. In order to get MFCC features, DCT has applied to the banks that gotten after STFT. Lastly, the mean normal is applied. The whole process given in the Figure 3.1.

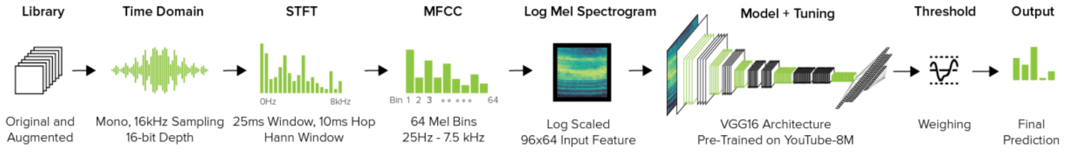

Library
Time Domain
STFT
MFCC
Log Mel Spectrogram
Model + Tuning
Threshold
Output
0Hz
8kHz
Bin 1 2 3 64
Original and Augmented
Mono, 16kHz Sampling 16-bit Depth
25ms Window, 10ms Hop Hann Window
64 Mel Bins 25Hz - 7.5 kHz
Log Scaled 96x64 Input Feature
VGG16 Architecture Pre-Trained on YouTube-8M
Weighing
Final Prediction

Figure 3.1 : Sound Classification Process Overview

3.1 Raw Time Domain of The Signals

As given in the Figure 3.1 16 bit wav file is taken called 17592-5-1-3.wav which is a running engine ate idle state. The sound samped in 16000 Hz as it be seen below:

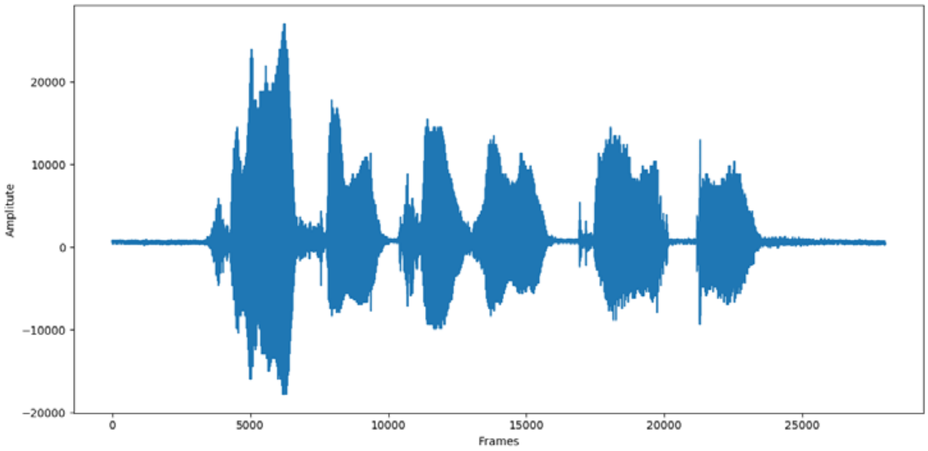


Figure 3.2 : Time domain of the signal taken from the UrbanSound8K Dataset

3.2 Pre-Emphasis

The first step to raw data is to implement a pre-emphasis on the signal to amplify the high frequencies. The pre-emphasis filter is useful in certain ways:

1. scale the frequency spectrum as high frequencies regularly have poorer magnitudes opposed to lower frequencies,
2. Avoid mathematical difficulties through the Fourier transform process and
3. Improve the Signal-to-Noise Ratio (SNR).

The filter can be employed to a signal $x(t)$ using the first order filter in the resulting equation:

$$y(t) = x(t) - \alpha x(t-1) \tag{3.1}$$

A typical value for the filter coefficient α is 0.95.

Pre-emphasis has a decent outcome in modern operations, largely because most of the motives for the pre-emphasis filter can be obtained applying mean normalization but for circumventing the Fourier transform problems which should not be a difficult in modern FFT applications.

After the implementation, the formula above the output becomes as in Figure 3.3:

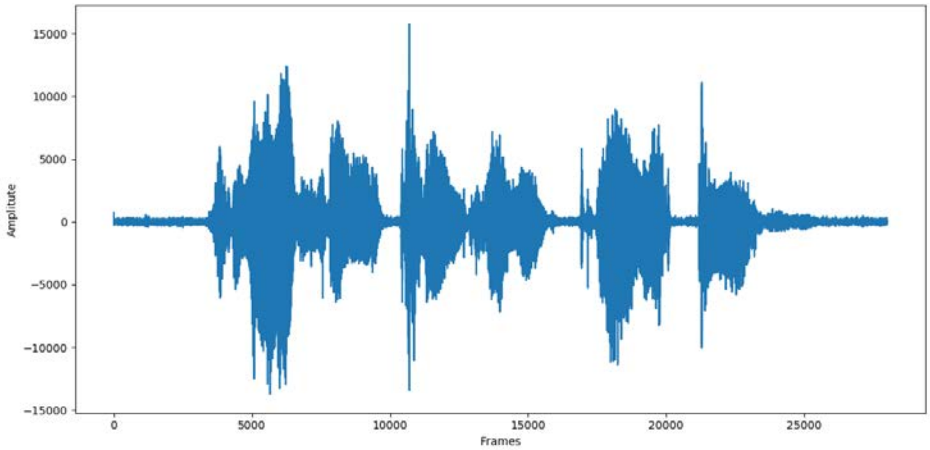


Figure 3.3 : Time domain with Pre-Emphasis Enchanment

3.3 Framing and Window

After the pre-emphasis step data needs to divide into short time packages. The idea behind this is to time dependency of the signal over time. In order to freeze information in a practical way, short time intervals created to form the one big chunk of sound information. It those do not make sense to apply FFT to the across the entire signal. With this approximation frequency, contours can be extracted in adjacent frames. As rule of thumb, 25 ms for frame size and 10 ms overlap is common usage.

After the division of the signal to short frames, a window filter is applied such as butter hamming window to each one of them. The Hamming window is given as

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \quad (3.2)$$

where, $0 \leq n \leq N-1$, N is the window length.

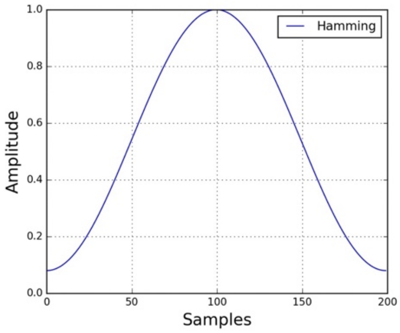


Figure 3.4 : Hamming Window

There are several reasons why it is needed to apply a window function to the frames, notably to prevent the acceptance made by the FFT that the data is infinite and to reduce spectral leakage shown above the Figure 3.4.

3.4 Power Specturm, Fourier Transform Filter Banks

N point FFT on every frame is applied to get STFT NFFT as 512 is taken is soffiecnt for most calculations. This step is consisting of filter thata are series of filters whisch are shaped as narrow triangles to extract frequency bands in Mel-scale. This scale aims to non liinear huam ear hearing perception of sound. Wihch rounds out to be not only applicable to huamn centric domins but also to dimension reduction as the Mel using the following equations:

The mel scale is a logarithmic frequency scale that tries to properly adapt to individual hearing. It was explained by studying with the human understanding of pitch 1940s with the single objective of defining the human auditory system on a linear scale[21]. An analysis revealed that the tone is linearly recognized in the frequency spectrum 0-1K Hz. Over 1K Hz, the scale shifts logarithmic. An estimated method widely used for Mel-scale has given here:

$$f_{mel} = 2595\ log_{10}\left[1 + \frac{f_{Hz}}{700}\right] \tag{3.3}$$

Each Mel filter targets a narrow peak of frequency, and on the sides of the filter the diminish rapidly. As shown in the Figure 3.5:

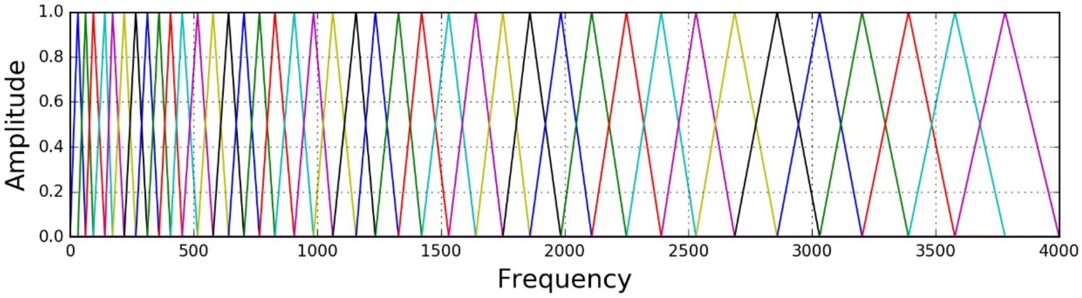


Figure 3.5 : Mel Scale Filter Banks.

After convulving of the filter bank to the power spectrum given in the Figure 3.6 the consecutive spectrogram is has gotten.

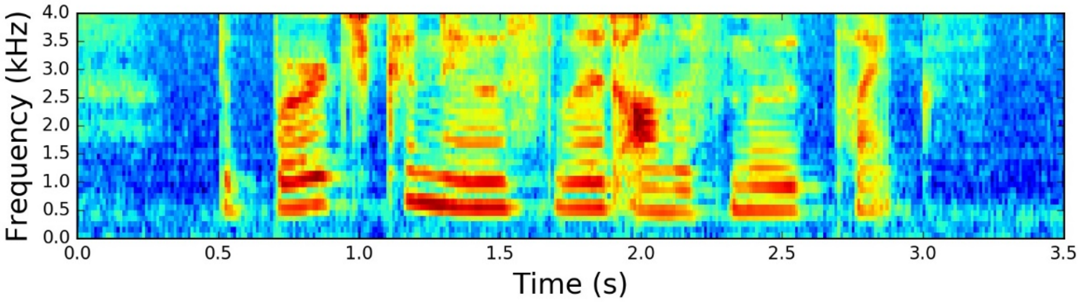


Figure 3.6 : Spectrogram of the Sound Signal

Lastly, Figure 3.7 shows the MFCC coefficients calculated from spectrum.

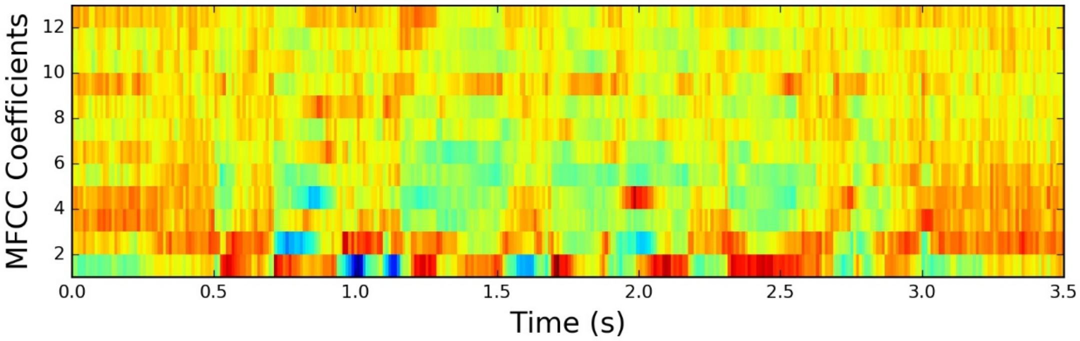


Figure 3.7 : The Resulting MFCCs

Turns out to be those filter bank coefficients calculated in the early are profoundly correlated, which could be problematic in any machine learning algorithms. Hence, a Discrete Cosine Transform (DCT) could be applied to curb correlation the filter bank coefficients and produce a compacted demonstration of the filter banks. Normally, for Automatic Speech Recognition (ASR), the resultant cepstral parameters are retained up to 14, and the rest have discarded. The reasons for rejecting the other factors is that they represent volatiles in the filter bank parameters and these fine details do not add to the recognition of impulsive or short-lived sounds.

After all, steps introduced for the filter bank and the MFCC. All the steps for the filter bank was a concern of human nature and promoted to the human perception of sound whereas MFCC steps are the concern of the machine learning algorithms limitations. DCT needs to be applied for decorrelation of signals. Which is called whitening. To eliminate and compensate with the power of Deep learning, MFCC selected instead of Mel-scaled filter banks.

3.5 Software

This section will describe the software implemented on the hardware described in Chapter 1. As a platform a Raspberry Pi is used.

All the programming was performed in Python, version 2.7. Besides all the general libraries used for data processing and analysis in Python such as Numpy or Mat-plotlib, four specific libraries were used in this project: The library SoundFile was used for reading and writing audio files.

The audio analysis library Librosa was used for the resampling of the audio files as well as for generating the Mel-frequency spectrograms that were fed as training data to the network. For the neural network programming part, Google's library Tensorflow was used, version 1.1.0. The library Scikit-learn used for calculating the confusion matrixes shown in Chapter 4, Results. All the code used in this project can be found in https://github.com/gyillikci/thesis.git.

3.5.1 Python

Python is an interpreted high-level programming language, object-oriented. İt uses dynamic semantic with high-level data structures and dynamic binding which makes it popular among researchers and the ones who look for rapid prototyping. Python is also an excellent tool to adhere to existing modules together. Python also supports a tremendous amount of modules and packages which makes very easy to use codes already written before.

3.5.2 Scikit

Scikit is software for machine learning developments for the Python programming language. There features such as classification, regression, Support vector machine, random forest, DBSCan, and other clustering algorithms. Scikit also incorporates two major libraries of Python’s namely numerical and scientific Numpy and scipy.

3.5.3 Micropython

Micropython approaches hardware control on a single board computer with an understanding of the object-oriented way by introducing modules functions and classes for abstraction. With the least amount of alternation Pin, UART, I2C and ADC functionalities can be controlled over the library.

3.5.4 LibRosa

Librosa is a package for audio analysis which utilizes essential tools necessary to get information retrieval functions. Standard tools are available such as DSP, Display, Feature extraction, Spectrum decomposition, temporal segmentation.

3.6 Convolutional Neural Network

Like other neural networks, they are made up of concevable wights and biases. Every neuron cell receives several inputs performs a dot product. The primary goal is to form raw image pixels to generate a score at the end side of the network. As to maximize the reliability fully connected the last layer involves an SVM.

The novel approach of the convents architecture makes the explicit assumption that the input is an image. By this assumption, certain limitations and constraints can be applied to the meta architecture of the network. This mechanism makes the forward function to work much more optimized.

Convolutional neural networks assume that the inputs are 2D fixed, sized and with a depth of 3 channels. With this knowledge, the encoding of specific imagery patterns or properties which fed into the network makes the forward function more robust towards generalization. This dimensional reduction leads to cost-cutting in both memory and CPU consumptions. In particular, CNN networks exploit two main approaches ideas

1. Sparse connectivity

2. Parameter sharing

Image have pixels in some millions yet, features that have searched for such as edges with kernels consumes only a fraction of the total number fed into the system as an image. In other words, filter kernels only use tens of pixels. This leads to very few parameters are required. For both reasons, it reduces the memory consumption of the model, and this commonality improves the statistical performance of the network shown in the Figure 3.8. This explains the principle of sparse connectivity.

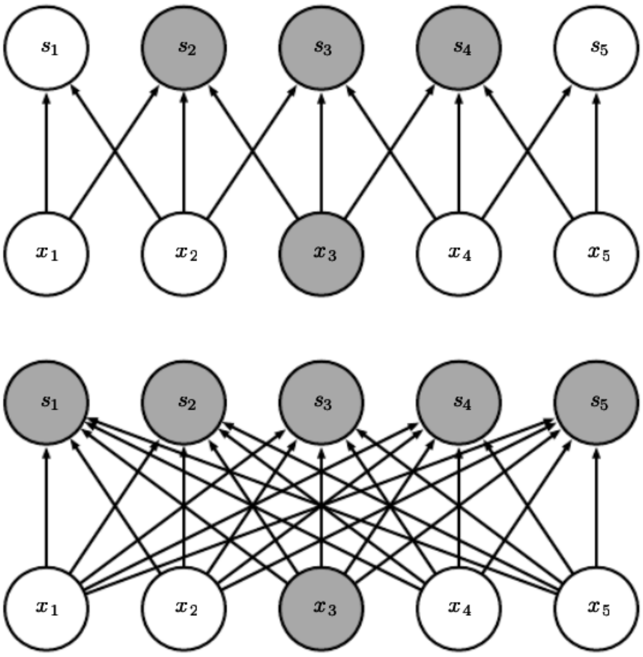


Figure 3.8 : Sparse connectivity, viewed from below. (Top) The selected unit is marked as x_3, and the output units s_2,3,4 that are a □ected by x_3. (Bottom) s_1,2,3,4,5 are a □ected by x_3 so, the relation is no long arse [15]

Parameter sharing indicates repeated use of the very same filters for multiple numbers of functions. The assumption is based on that features would be useful to detect some spatial proper more than one place on the image. The very same mechanism can be applied to another spatial position on the image. In fully connected neural networks, each node of the weight matrix tends to learn very similar patterns again and again which leads to exponential growth of resource consumption. As a result, the computational burden becomes too heavy to apply these kinds of networks to use in image-based problems. Instead, it is used precisely once while calculating the output of a layer. It has multiplied by one node of the input and then never returned. In a CNN, each member of the kernel is used at every position of the input which is given in the Figure 3.9. The parameter sharing enables not to learn parameters for every location; instead, the network learns the only one set.

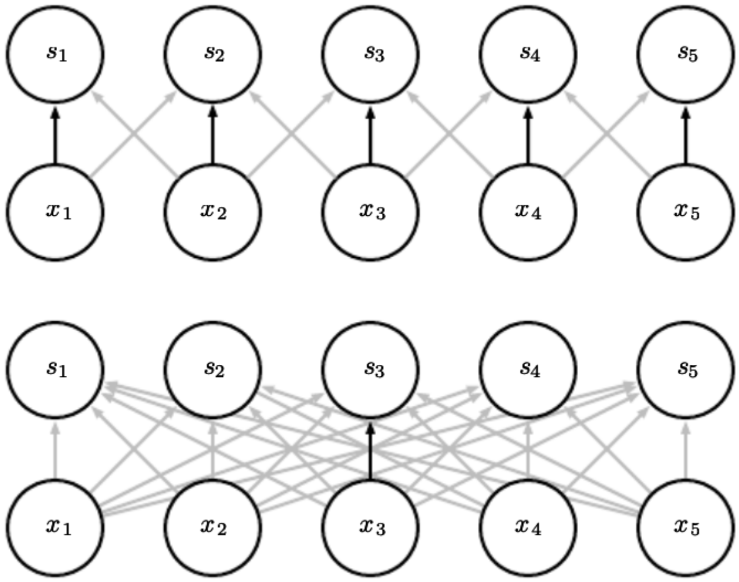


Figure 3.9 : Parameter sharing[15]

Parameter sharing. Black pointers show the connections that use a particular parameter in two different models. (Top) The black darts show uses of the pivot element of a 3-element kernel in a convolution. The Nature of parameter sharing, the single parameter is used at all input locations. (Bottom) The single black arrow indicates the use of the central element of the weight matrix in a fully connected model. This model has no parameter sharing, so the parameter is used only once. [15]

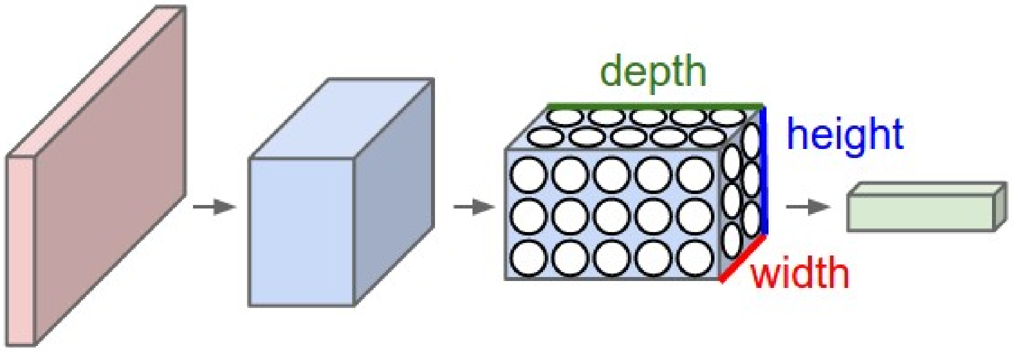


Figure 3.10 : CNN manages its neurons in 3D (depth, width, height,), as imagined in one of the layers. Each layer of a CNN reconstructs the 3D input volume to a 3D output-volume of neuron activations[10]

3.6.1 Overview of architecture

Fully connected neural networks do not scale close to linearity. For instance, images that have a dimension of 32*32*3 would have 32*32*3 = 3072 weight. For the sake of this small example still manageable.

As it can be seen in the Figure 3.10 when this kind of approach has taken to be applied to 200*200*3 dimensions results as 120000 weights. The fully connected structure does not scale in a well-behaved manner. Furthermore, as the fully connected structures have many more neurons can cause some drawbacks such as overfitting and wasteful unnecessary connections.

The volume of neurons of the convolutional neural networks takes advantage of the fact that the input is an image width, height, and depth. Here the depth corresponds to the channels of an image but not to the total length of the network that is in use.

In particular, CNN which can be seen in the Figure 3.11 is a stack of layers. One-layer feeds to the other layer coming after. Every layer transform 3D volume to an output of transformed 3D volume with some alternations on it.

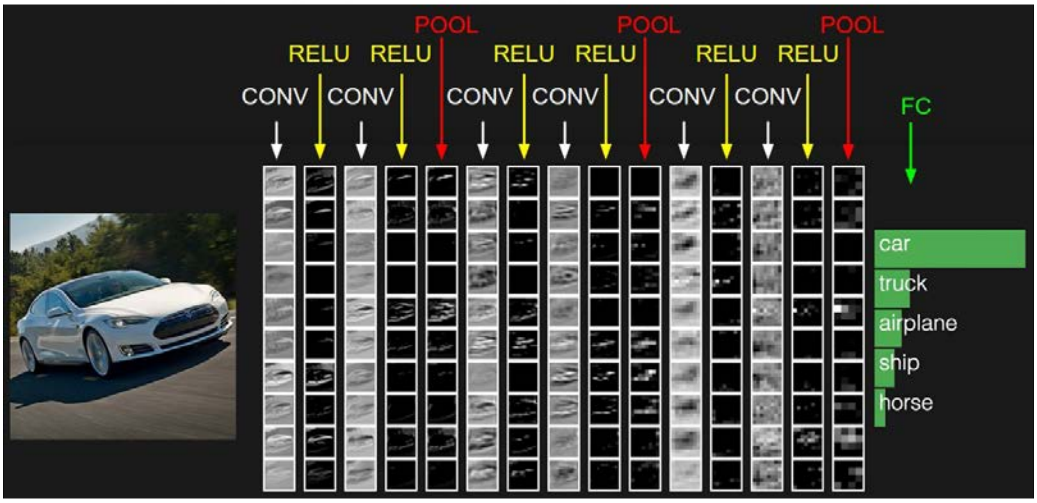


Figure 3.11 : An illustration ConvNet architecture[10]

An illustration ConvNet architecture. The primary volume stores the raw image pixels at left and the latest volume reserves the class scores at right. Every volume of activations with the processing path is shown as a column. As it is hard to conceive 3D volumes, the extended each slices of volume in rows. The last layer keeps the rates for each class. However only top 5 scores are visualized [10]

3.6.2 Transfer learning

Transfer Learning is deep learning approach that exploits prior knowledge information from a trained dataset. It is used as a starting line for the newt model to be subject to training. Traditional machine learning diverges with transfer learning in the sense of pre-trained model which has been explored for another task to jump-start the development process a new problem. It involves the idea of a domain. The positive side to using Transfer learning is the reason can accelerate time to train big data models from ground zero instead. By reusing the pre-trained model as a starting point speeds training model process.

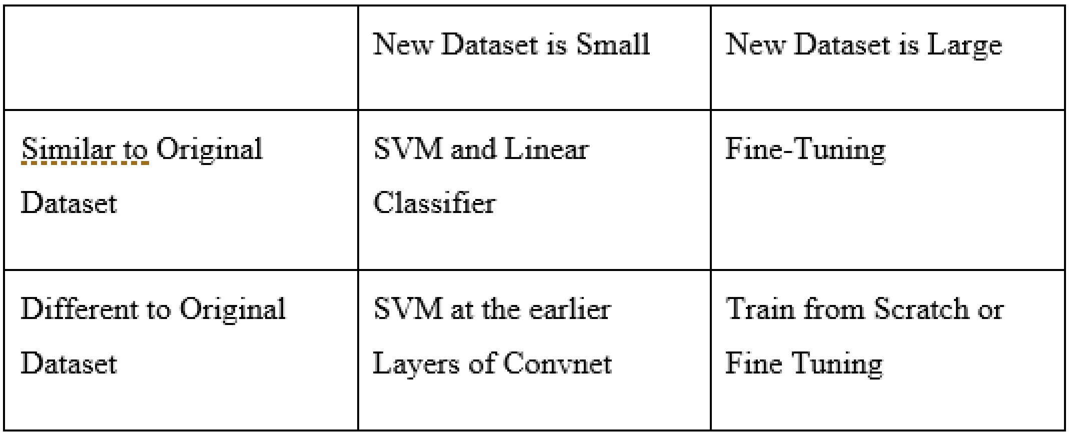
In the practical way very small percentage train an entire CNN from the ground. In other words, random initialization. The reason is the amount of dataset needed to do it. It is the common way to use pertained model and then use it as fixed future extraction for the task of interest. There are mainly three majör transfer learning approaches:

1) Pretraind models: CovNets take a couple of weeks for a successive training. Across multiple GPUs. For the benefit to other researchers submit their checkpoints. Fort he sake of other researchers who can use these checkpoints as a starting point for their fine-tuning.

2) CNNs as Fixed feature Extractor: By removing last fully connected layer of a CNN. Then threat the rest of the CNN as a fixed feature extractor fort he new dataset. For example, in Alexnet this would correspond to 4096 vectors for every image that consists activation of the hidden layer just before the classifier. It is called CNN codes. For the reuse process, these codes are applied to RELU in order to make them thresholded at zero. After extraction of the codes for all images training a linear classifier such as SVM for the new dataset has happened

3) Fine- tune the CNN: the last strategy is to both replace the last fully connected layer but also resume to train the CNN with the new dataset for the sake of preventing overfitting and continue to make some more backpropagation. Some previous layers such as color blob detection, edge detector. Some other latter layers of the CNN subject to more complex and specific details of the classes contains fort he dataset images. When and how to use fine-tune: this is an equation which has

multiple variables and factors. Most important of the first to factors are the size of the dataset and the similarity to the dataset. CNNs features are mostly common on the earlier layers. The rule of thumb approach can be summarized as the representation given in the Figure 3.12.



	New Dataset is Small	New Dataset is Large
Similar to Original Dataset	SVM and Linear Classifier	Fine-Tuning
Different to Original Dataset	SVM at the earlier Layers of Convnet	Train from Scratch or Fine Tuning

Figure 3.12 : Fine Tunning Rule of Thumb Representation

4. OBJECT DETECTION

In modern computer vision, object detection is studied collectively with the object classification. Localization, tracking, and extracting information from the object are the stack upon classification respectively. These processes are highly correlated on object detection. In taxonomy, the purpose is to determine the object class. In a broad sense, localization locates the location of the objects. The tracking process in a video is an approach to get the movement by inter-frame relation of the localized object. The purpose of an object detection operation is to name and find the location of all objects. The input for the detector is an image containing the object, and the output is a list of the bounding box. Figure 4.1 below object classification, detection, and localization.

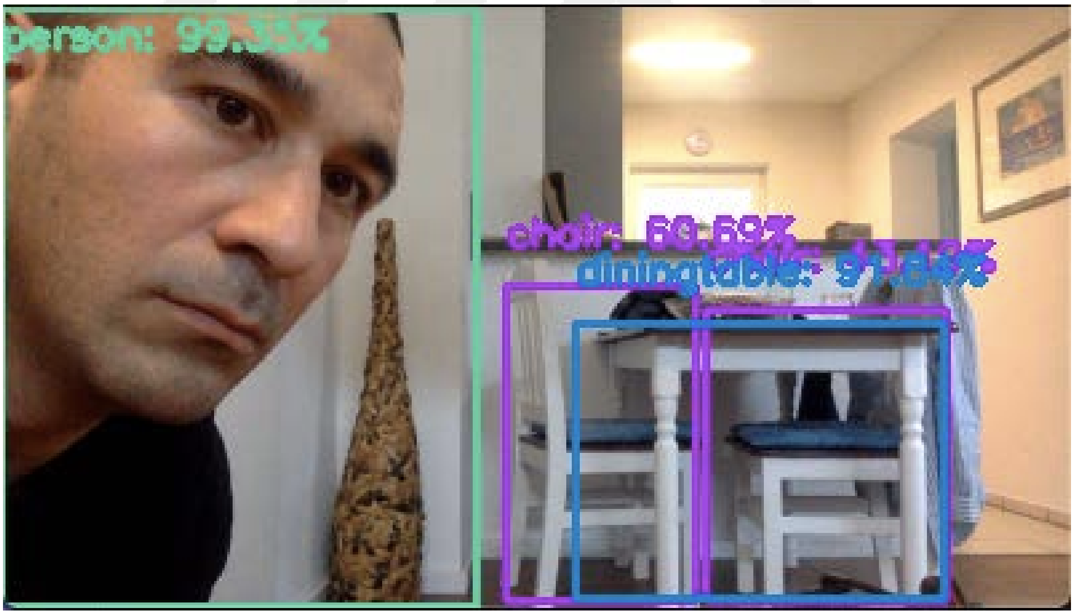


Figure 4.1 : Prototype Object Detected with a trained model of MobileNetSSD

Object detection was based on obtaining the feature descriptors from the images before the deep learning approaches became common. Histogram Oriented Gradient (HOG) and Scale Invariant Feature Transform (SIFT) [20] including Support Vector Machines (SVMs) classification were the principal methods for object detection. Deep CNN based models exceed the classical object detection models. Much robust Region

Based Convolutional Neural Network (RCNN)[12], Region-Based Fully Convolutional Network (RFCN) [19] and single shot detector (SSD) are now principal architectures in object detection models. Formentined architectures contain the deep CNN together with the feature extractors. Inception, MobileNet, ResNet and VGG are common and accepted feature extractors. These extractors can be used in meta-architectures as stated earlier depending on application type. Object detection is the trending area in machine learning. Popular area of object detection can be given as facial landmark detection, object counting, image search, landmark identification, satellite image analysis, autonomous driving cars, commercial and military drone. Detection, Dlib, RetinaNet, TensorFlow and You only look for Once (YOLO) are current libraries that are used for the studies. Deep CNN frameworks are implemented in all modern object detection libraries.

4.1 Single Shot Detector

A single shot multi-box detector (SSD) meta-architecture solves the object detection problem by using a single pass of a feedforward convolutional network. Unlike other models, this does not generate region proposals nor do resampling of image-segments thus saving computational time [9, 31]. This network handles objects of different sizes by using features maps from different convolutional layers as input to the classifier. This network produces a large number of bounding boxes with the scores of an object class in those boxes.

Non-maximum suppression is used to eliminate boxes below a certain threshold so that only the boxes with higher confidence values proceed for classification. SSD meta-architecture allows end-to-end training to and improving the speed of the detector. This meta-architecture does everything in one shot. Thus, it is faster than other meta-architectures, but it lags the detection accuracy. The whole process is shown in the Figure 4.2 below.

The SSD layer architecture is built on top of a feedforward CNN that results in a fixed-size collection of bounding boxes and object class instances present in those boxes. The input image is passed over a group of convolutional layers and downsampled through the SSD layer shown in the figure below.

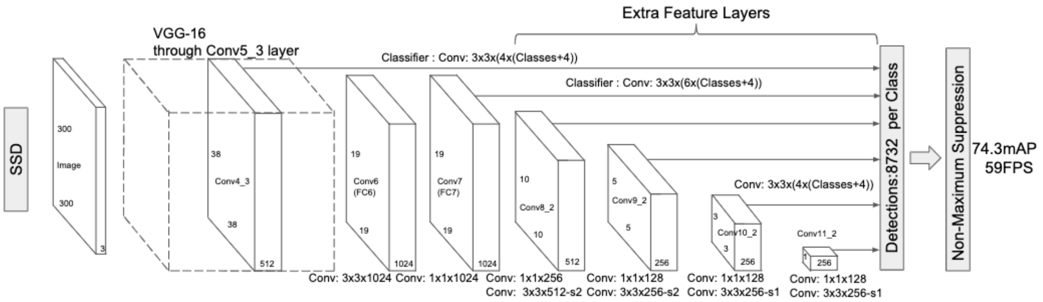


Figure 4.2 : The SSD model concatinate several feature layers folowing the base network

This SSD layer is linked to the output of the last convolutional layer of the root model. Reoccurring collections of feature maps at different scales are obtained from the convolutional layers with the prediction of object classes. The predicted boxes are compared with the ground truth of the object, and the best one with higher IoU is selected concurrently with the higher probability score from class predictor.

4.2 Feature Extraction

The feature extractor is the main building section of the detection model that is used to derive the features of objects from the data. The detection model composition is composed of detection architecture, feature extractor and classifiers shown in the figure. The input image is passed over the extractor that selects features from the picture. The selected elements are forwarded to the classifier that analyzes the class and the location of the object in the input image as represented in the Figure 4.3.

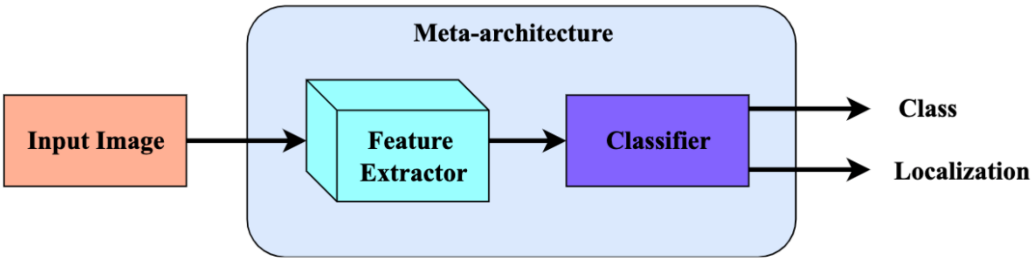


Figure 4.3 : Object Detection model Architecture for Classification and Feature Extraction

The feature extractor is an architecture that aims to increase accuracy while decreasing the computational complexity. MobileNet, AlexNet, NAS, ResNet, Inception, and

VGGare, some common extractors that can be performed in the detection meta-architectures.

4.3 Mobile Network

MobileNet which is given in the Figure 4.4 is a deep neural network that is suitable for mobile and embedded devices with limited resources. The aim behind this architecture is the separation of the standard convolutional filter into two separate convolution filters, namely depthwise and pointwise convolution. The calculation complexity of the standard convolutional filter is higher than the aforementioned separated filters computation complexity of convolutions.

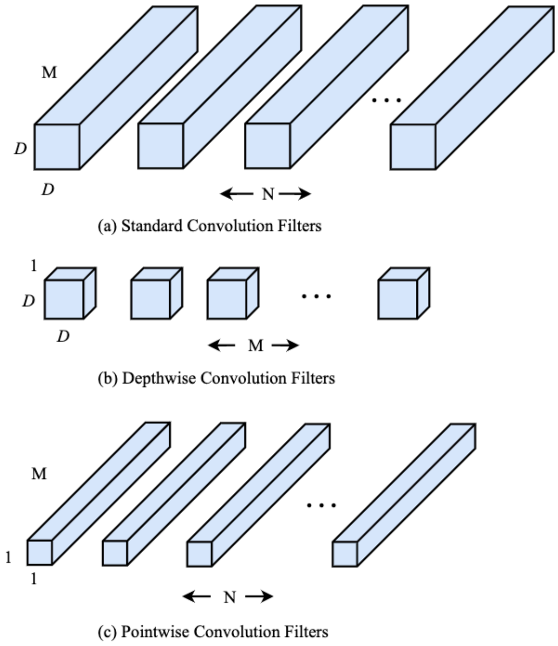

M
D
D
N
(a) Standard Convolution Filters
1
D
D
M
(b) Depthwise Convolution Filters
M
1
1
N
(c) Pointwise Convolution Filters

Figure 4.4 : Prototype MobileNet feature extraction based on the separation of the standard convolutional filters into depthwise conv and pointwise conv

The computation cost of the convolution depends on the data M, size of the output network N, feature map size and the kernel size. The computation complexity of the standard convolutional filter shown in Figure above is larger than the total calculated cost of the depthwise (b) and the pointwise (c) convolution filters. This division is optimized for the calculation rate. The decrease in accuracy is rather small in ratio. These are the variables for tuning between rate and precision.

4.4 Tensorflow Object Detection

TensorFlow is an open-source machine learning library published by Google [17]. Google leading developers ingeniously developed it to convey AI research and implemented Google services like speech recognition, mail analytics, photos, and pictures. The Tensorflow framework is implemented in C++ and has a solid Python wrapper.

It became a popular library among deep learning developers and researcher just after it was released to the public. [16] TensorFlow library the statistical calculations performed with data stream graphs. Nodes express mathematical operations and the edges denote the multidimensional arrays additionally identified as tensors. These tensors are exchanged among sides.

TensorBoard is a graphical user environment for the TensorBoard. This GUI demonstrates the progress of the training performance of a model. a flowchart of the data transformations, visualize summary logs over time, traces the performance of algorithm and computation graph before and during runtime.

The exciting thing in TensorFlow library is that computations are expressed as flowcharts, separating design from implementation. The same model can be implemented on powerful computing devices with large numbers of processors and mobile devices. TensorFlow provides a wide range of functions and classes that allow users to build the desired models from scratch [16].

5. PROTOTYPING AND MEASUREMENTS

The Prototype is build up of a camera, Macbook Air, microphone array, Raspberry Pi, robot servo, and power for the electronics. As can be seen in the figure below, Camera and the servo is mounted to the Raspberry Pi's USB and GPIO 12 respectively. Microphone Array related pins are connected directly to the 40 pinouts of the Raspberry Pi. For the sake of computatinal power which Raspberry Pi cannot support due to its limitation a Macbook Air with 8Gb RAM is used for real time object detection. Data exchange between Macbook Air and the Raspberry Pi performed by a simple local client-server process. The prototype can be seen in the Figure 5.1 below.

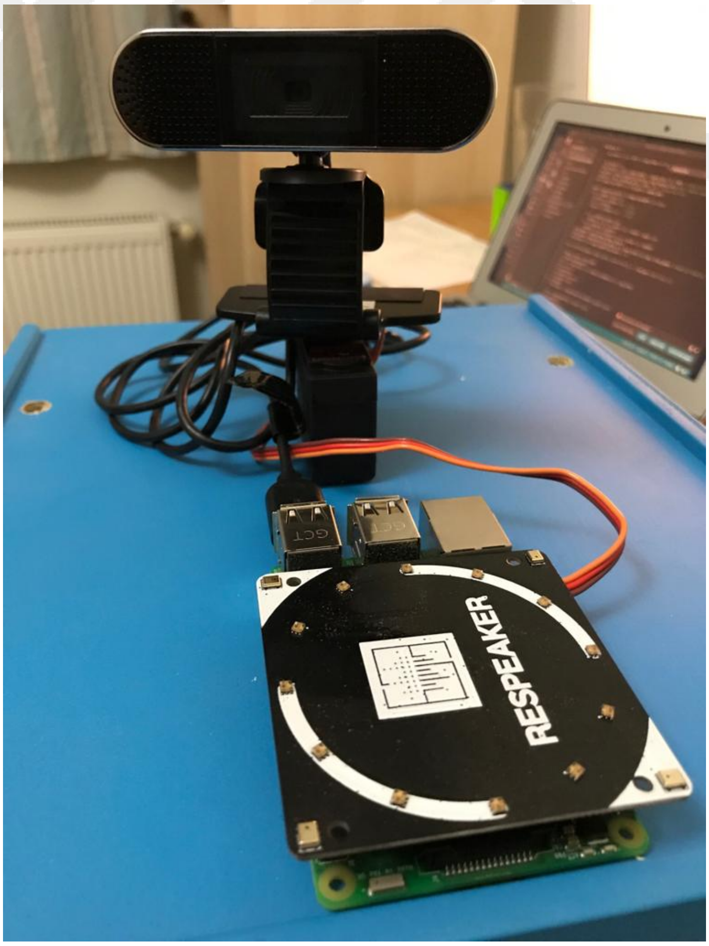


Figure 5.1 : Prototype Setup consist of a camera, Macbook Air, robot servo, Microphone array and a Raspberr Pi

5.1 Microphone Array

The given in the Figure 5.2 and Figure 5.3 is for Raspberry Pi with 4-microphone that is desined to work for voice applications. One can build versatile voice realted applications which can be act as edge device or can feed stream to a cloud infrastructure for computations. Main features as seen below:

- Rapberry Pi compatible
- 4 microphones
- Indoor voice capture capability up to 10 feets

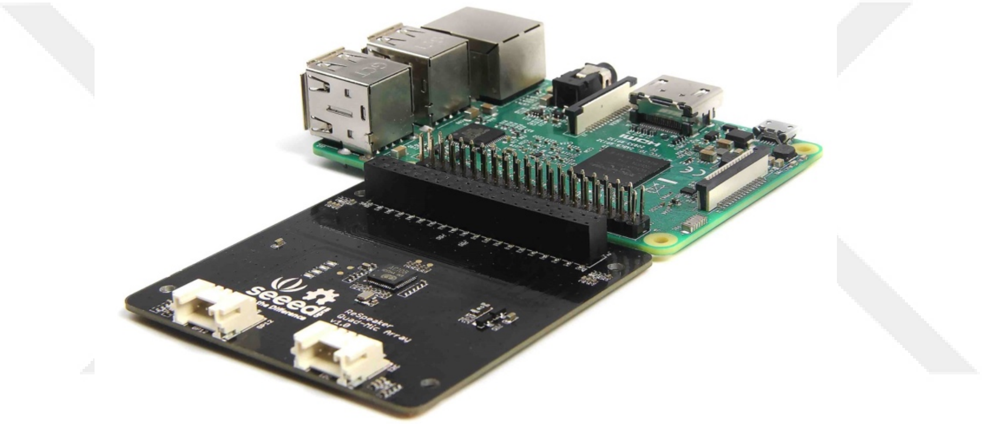


Figure 5.2 : Raspberry Pi and ReSpeaker 4-Mic Array

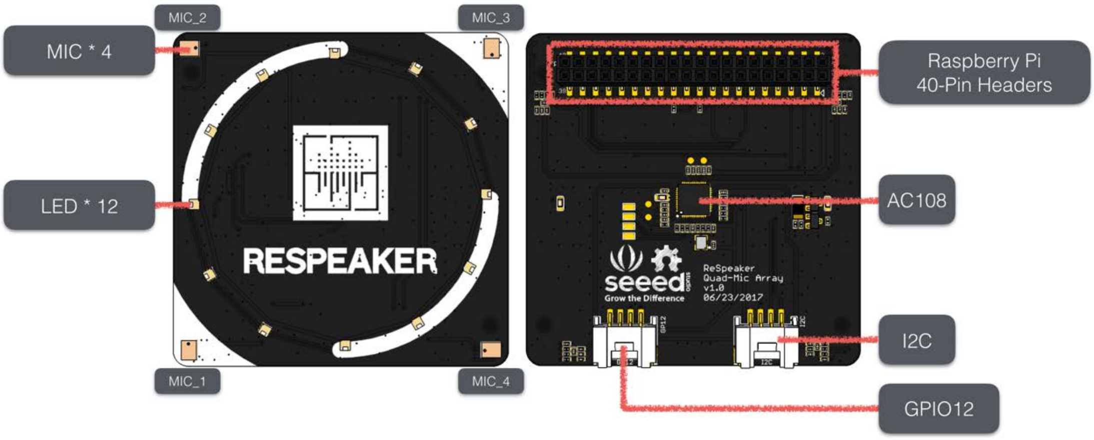


Figure 5.3 : Detailed Illustration of the Microphone Array

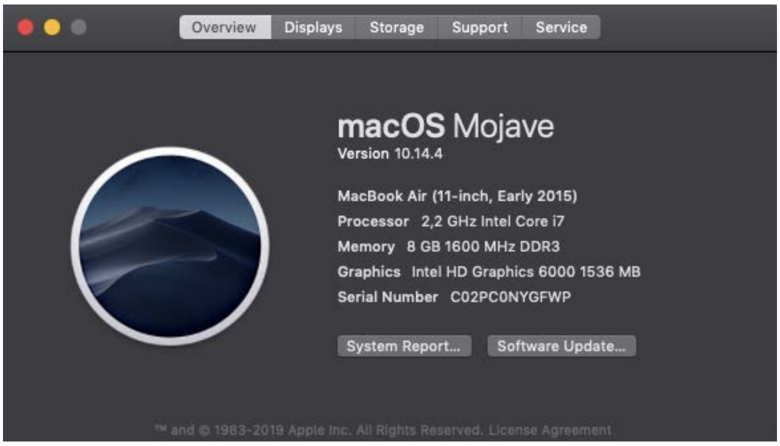
5.2 Camera

The Full HD Pro Webcam which can be seen in the Figure 5.4 from Fujitsu gives excellent quality video capture with a smart design that clips easily onto your notebook or monitor. Fine 1080p video and 8MP still image abilities provide a lifelike picture for your recordings and video calls. The device is Plug and Playable and has auto-focus built-in so that you can be started with your Full HD video capture in minutes.

Figure 5.4 : Webcam for the Prototyping

5.3 Macbook Air

The MacBook Air is a laptop developed and produced by Apple Inc. This consists of a laptop standard keyboard, an aluminum case, and a thin light housing. This MacBook Air configuration has solid-state drive storage and Intel Core i7 CPUs as it can be seen in the Figure 5.5 below.



macOS Mojave
Version 10.14.4

MacBook Air (11-inch, Early 2015)
Processor 2,2 GHz Intel Core i7
Memory 8 GB 1600 MHz DDR3
Graphics Intel HD Graphics 6000 1536 MB
Serial Number C02PC0NYGFWP

™ and © 1983-2019 Apple Inc. All Rights Reserved. License Agreement

Figure 5.5 : Macbook Air Configuration for the Prototyping

5.4 Servo Motor PID Controller

A general feedback control loop called a Proportional Integral Derivative (PID) controller shown in the Figure 5.6. PID in general used in motorization such that an actuator can reach an optimum value. All feeding process observed by a quickly and correctly. PIDs are widely used in robotics. The Proportional controller calculates the present error term, the difference between the desired setpoint and sensor reading and has a goal of compensating for the error.

The integral controller holds the small errors that are cumulated over time. Lastly, the Derivative controller calculates the error would be caused by the present speed or action. The harmony of the three would give a smooth and robust flow of an actuator.

The sensor output is known as the “process variable” and serves as input to the equation. In every stage the feedback loop, timing is taken, and it is input to the equation as well.

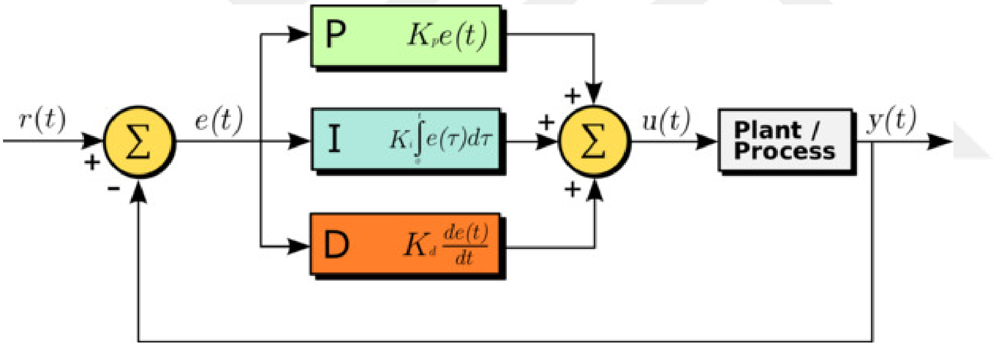


Figure 5.6 : Diagram of a PID controller

The used servo model is an hobby servo motor which is good enough to carry torque induced by the mass of the webcam mounted on the top of the servo shown in the Figure 5.7 below.

Figure 5.7 : Servo Motor

5.5 Sound Measurements

For the measurement, an Angular scale shown in the Figure 5.8 for degrees is located under the prototype platform for to determine ground truth angle for the sound and visual localization. As seen in the below figure, the prototyping platform centered on the angular scale.

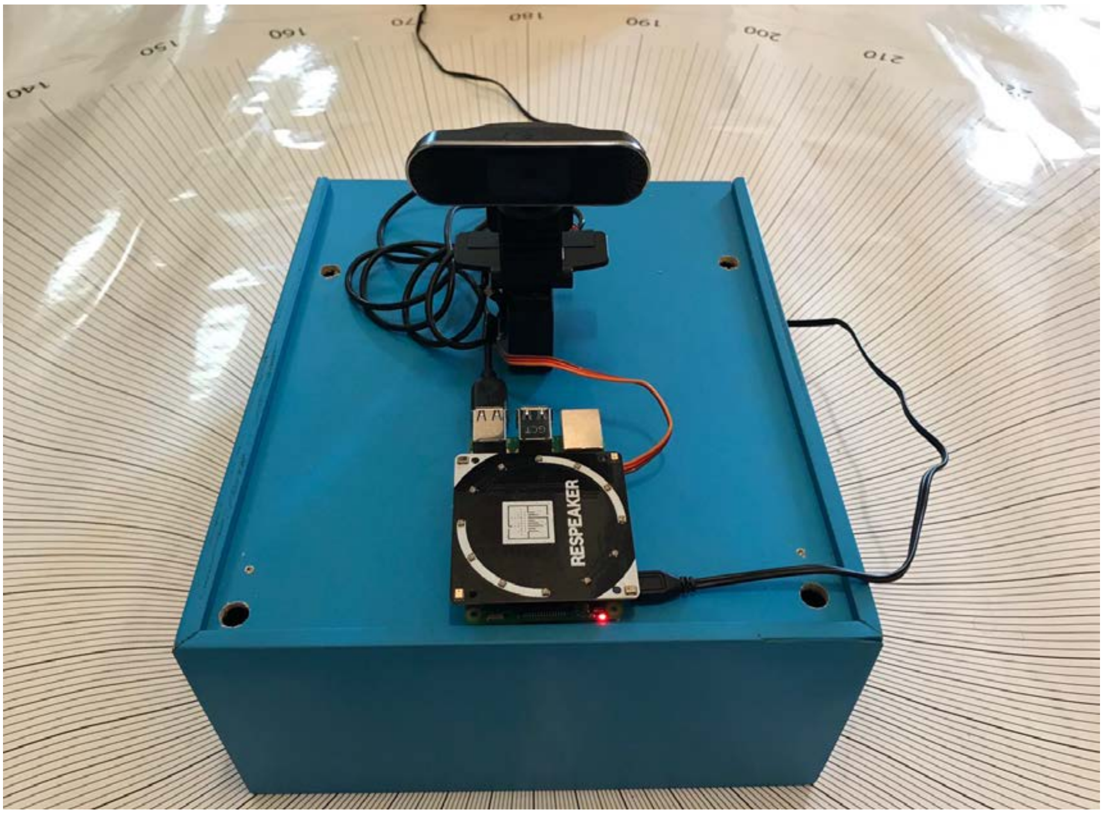


Figure 5.8 : Angle Scale for the Setup Platform

For the practical purposes of this thesis accuracy of the sound source localization was more than enough to locate sound source. The error occurred on the sound source measurements were around ±2 degrees. This would be highly correlated with the measurement errors, echo of sounds in the room. Nevertheless, Sound source localization worked as expected and gave the high accuracy angle information for the visual object detection where to look for.

5.6 Sound Source Dataset

After locating any sound source, sound classification is the second step to determine what kind of sound is generated. For this purpose, neural network is used, it is made up the way to classify urban sounds into classes using machine learning will be used. There are fundamental problems to classify data where it is in a vector form. Even though for the textual dataset and anomaly detection dataset tf-ids can be used but it comes to sound classification its effectiveness fades away. Sound classification feature extraction by itself is not enough. On the other hand, feature extraction is the starting point for sound classification.

5.6.1 Urban sound dataset

For the sound classification process sound, oriented data set is required Researchers published urban sound dataset which consists of 10 folds of directories with total number of 8732 labeled sounds. Transformation from time domain to logarithmic spectrogram shown in the Figure 5.9. Clips are 4000 milliseconds in general. The classes are as follows: air conditioner, drilling, engine idling, gunshot, jackhammer, siren, car horn, children playing, dog bark, street music sound files are stored as .wav format.

Visualization of the files are as follows:

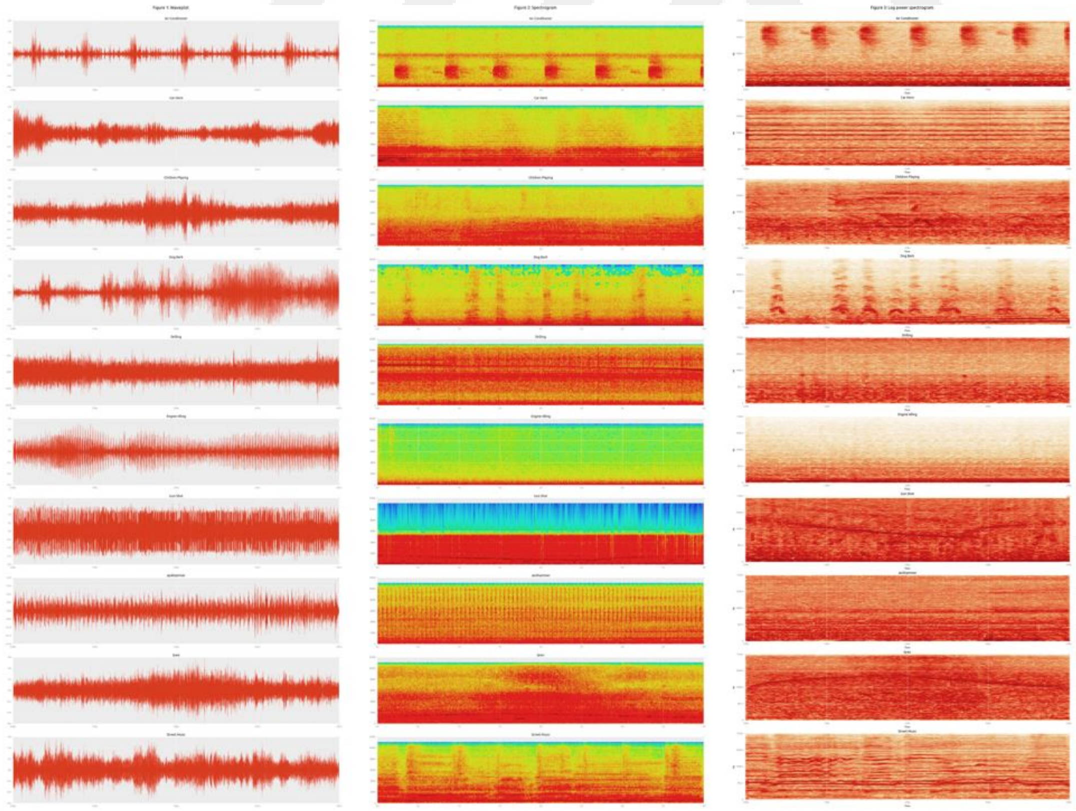


Figure 5.9 : An illustration of air conditioner, drilling, engine idling, gunshot, jackhammer, siren, car horn, children playing, dog bark, street music in Time Domain, as Spectrograms, as Log Scaled Spectrograms

As mentioned above, there are 10 different sounds from the urban areas. Before the training procedure. Preprocessing is applied in order to have sound information as an imag

5.7 Training Procedure for Sound Classificiation

This section discusses the retraining of the pretrained MobileNet[11]. It is an efficient Convolutional Neural Network. The MobileNet model is configurable in 2 ways.

1) input resolution can be 128, 160,192 or 224px. Higher numbers can be fed to the system for training, yet it would cost more processing power. As a result, classification accuracy, would be better.

2) The relative sizing of the MobileNet model can be scaled with the multipliers of 1.0, 0,75, 0,50 and 0,25

For this study, the image size has taken as 224, and the scaling has taken as 0.5.

The number of computations required, and the accuracy taken with the size of the model (circular area). 16 circles have represented for the MobileNet. For each model size, there is a corresponding image resolution setting.

After the bottleneck files, have generated for the dataset, the actual training process starts only for the final layer of the network. For the training 16000 steps has taken and each batch uses ten randomly selected training set. For this process bottlenecks of the corresponding pictures are extracted from the cache and fed into the final layer for prediction making. Then, predictions have compared with the actual labels of the ten images. The resulting comparison is used to update weights of the final layer by executing backpropagation process.

As the training process resumes, a series of outcomes have collected such as accuracy, validation, and cross entropy:

Training accuracy indicates the percentage of the correctly labeled imaged in the current batch of images. Validation accuracy is the key to the precision of randomly picked images from an isolated set of images. Cross-entropy is the loss function which indicates how successful is the process is going on. The lower numbers are, the better.

The Figure 5.10 below indicates the progress of the model’s accuracy over the training process as it trains.

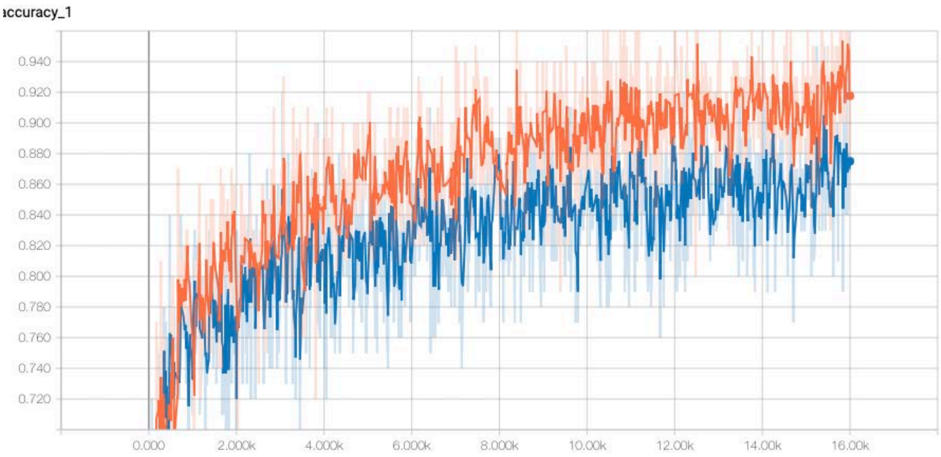


Figure 5.10 : An Training Accuracy of the UrbanSound8K for sound classification

The orange line in the Figure 5.11 indicates the accuracy for the model and the magenta line shows accuracy for the test images which were separate from training data images not to have overfitting. Overfitting can be explained as the model start to memorize the pictures instead of understanding the general patterns in the pictures. Without falling to the overfitting pitfall, the training performance continuously rises.

The trained model over MobileNet meta-architecture. As it can be seen, most of the classifications gave satisfactory classifications. This level of classification is not enough for an end product yet; it is enough for the integrity of the prototype purpose.

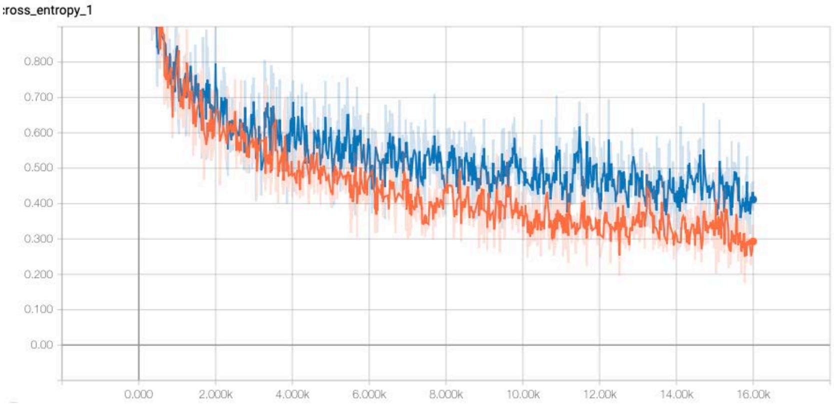
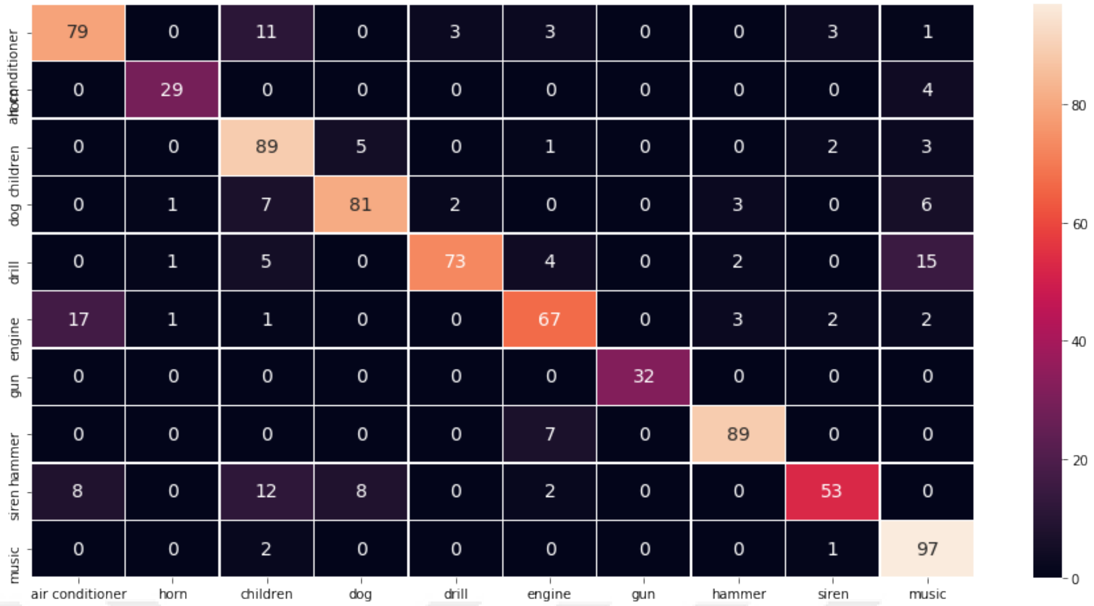


Figure 5.11 : Cross Entropy of the Training Procedure of the Urban8K for sound classification



	air conditioner	horn	children	dog	drill	engine	gun	hammer	siren	music
air conditioner	79	0	11	0	3	3	0	0	3	1
horn	0	29	0	0	0	0	0	0	0	4
children	0	0	89	5	0	1	0	0	2	3
dog	0	1	7	81	2	0	0	3	0	6
drill	0	1	5	0	73	4	0	2	0	15
engine	17	1	1	0	0	67	0	3	2	2
gun	0	0	0	0	0	0	32	0	0	0
hammer	0	0	0	0	0	7	0	89	0	0
siren	8	0	12	8	0	2	0	0	53	0
music	0	0	2	0	0	0	0	0	1	97

Figure 5.12 : The error matrix for the Urban Sound Dataset with trained model

The Figure 5.12 above shows the error matrix created againt to trained model and test samples shows stisfactory results for realtime performance.

After all sound related calculation and computations, the camera rotated to the angle determined from the multiple microphones. After the rotation towards the source, identified sound class of the particular sound classification is send to object detection unit to locate the sound source visually. Object detection unit of the prototype makes the relation with the sound source with the light of the classification information.

The framework has shown its capability to establish audio-visual integration and come up with a result of an object of the sound source even the the object is out of sight of the camera as shown in the Figure5.13.

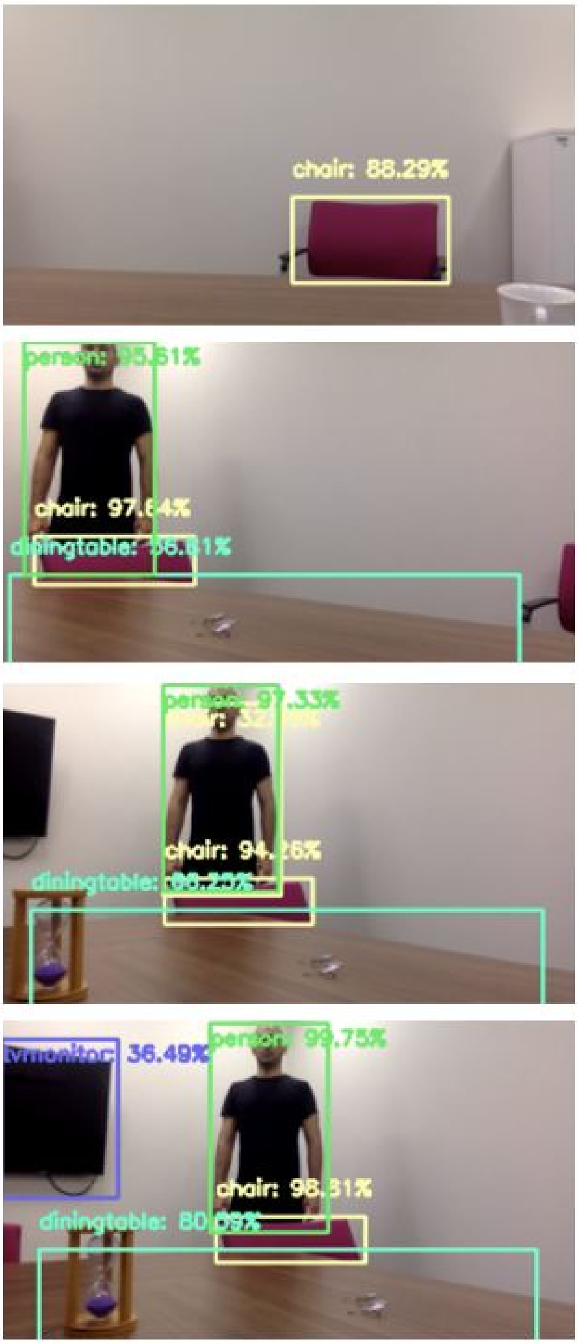


Figure 5.13 : The association of audio and visual information by first sound detection and rotating camera to the visual of the sound source

6. CONCLUSIONS AND RECOMMENDATIONS

The Audio-Visual platform consists of trained model with 10-fold UrbanSound8k for sound classification and pre trained MobileNet object detection with the integration of both models for associating the sound source with visual feedback. In both approaches neural networks are used to based on the fact that both performance and computational load Accuracy. Overall accuracy tends to slightly increase as the training steps increase yet it is needed to approach with the other method to achieve better results.

The thesis shows that the audio-visual platform can unlock real-time activity recognition by leveraging two of the most common sensors found in consumer electronics today. Namely, microphones and cameras. Mimicking the nature in terms of integrating two different sensors increases the awareness established exponentially with the least amount of price.

It is worth to mention that all components used in the prototype are off-the-shell meaning that it is easy to replicate and modify the protoype. This audio-visual integrated platform makes the awarness of the medium closer to living being like perception. By leveraging existing state-of-the-art sound classification and object detection models and tuning them with the audio-visual approach, the study enabled a general-purpose and flexible sound recognition-object detection pipeline that requires no prior adjustment.

REFERENCES

[1] **J. Salamon** (2017). “Deep convolutional neural networks and data augmentation for environmental sound classification”. IEEE Signal Processing Letters 24.3 pp. 279–283.

[2] **K. Piczak** (2015). “Environmental sound classification with convolutional neuralnetworks”. Machine Learning for Signal Processing (MLSP), 2015 IEEE25th International Workshop on. IEEE. pp. 1–6

[3] **H. Phan** (2017). “Audio scene classification with deep recurrent neural net-works”. arXiv preprint arXiv:1703.04770.

[4] **Y. Zeng** (2017). “Spectrogram based multi-task audio classification”. Multimedia Tools and Applications pp. 1–18.

[5] **Velasco, J.F., Taghizadeh, M.J., Asaei, A., Bourlard, H., Martín-Arguedas, C.J., Guarasa, J.M., & Pizarro-Perez, D.** (2015). Novel GCC-PHAT model in diffuse sound field for microphone array pairwise distance based calibration. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2669-2673.

[6] **Grondin, F., & Michaud, F.** (2019). Lightweight and Optimized Sound Source Localization and Tracking Methods for Open and Closed Microphone Array Configurations. CoRR, abs/1812.00115.

[7] **X. Huang, A. Acero, and H. Hon.** (2001). “Spoken Language Processing: A guide to theory, algorithm, and system development”. Prentice Hall

[8] **Brandstein, M. S. and Silverman, H. F.** (1997) “A robust method for speech signal time-delay estimation in reverberant rooms”, Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Munich, Germany.

[9] **Davis, S. Mermelstein, P.** (1980). “Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences”. IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 28 No. 4, pp. 357-366

[10] Convolutional Neural Networks for Visual Recognition. http://cs231n .github.io/convolutional-networks/.

[11] **Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H.** (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. CoRR, abs/1704.04861.

[12] **Lawrence, S., Giles, C.L., Tsoi, A.C., & Back, A.D.** (1997). Face recognition: a convolutional neural-network approach. IEEE transactions on neural networks, 8 1, 98-113.

[13] **Long, J., Shelhamer, E., & Darrell, T.** (2015). Fully convolutional networks for semantic segmentation. CVPR.

[14] **Url-2 < https://goo.gl/images/W67BkG** >, date retrieved 20.2.2019

[15] **Heaton, J.** (2017). Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning. Genetic Programming and Evolvable Machines, 19, 305-307.

[16] **Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P.A., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., & Zhang, X.** (2016). TensorFlow: A System for Large-Scale Machine Learning. OSDI.

[17] **Url-5** <*https://www.tensorflow.org/* >, 2019 G. B. Team. About tensorflow.

[18] **Zhang, X., Zhou, X., Lin, M., & Sun, J.** (2018). ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6848-6856.

[19] **Wang, L., Ouyang, W., Wang, X., & Lu, H.** (2015). Visual Tracking with Fully Convolutional Networks. 2015 IEEE International Conference on Computer Vision (ICCV), 3119-3127.

[20] **Cheung, W.A., & Hamarneh, G.** (2009). - SIFT :-Dimensional Scale Invariant Feature Transform.

[21] **Bamberger, R.H., & Smith, M.J.** (1992). A filter bank for the directional decomposition of images: theory and design. IEEE Trans. Signal Processing, 40, 882-893.

[22] **Wang, X., Girshick, R.B., Gupta, A., & He, K**. (2018). Non-local Neural Networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7794-7803.

CURRICULUM VITAE

Name, Surname	**:**	Giray YILLIKCI
Place and date of birth	**:**	Atlanta, Georgia. ABD
Permanent Address	**:**	Papatya sokak no: 6 Uskumrukoy, Sariyer Istanbul TURKIYE
E-Mail	**:**	gyillikci@gmail.com
B.Sc.	**:**	Physics, Istanbul Koc University