People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

Amar Thelidji University - Laghouat

FACULTY OF TECHNOLOGY

ELECTRONICS DEPARTMENT

Master's Dissertation

Prepared by: SEFARI Ali Elhocine / Telecomunication systems

KHELIFI Mohammed Elbachir / Telecommunication Networking

DOMAIN: Sciences and Technology

FILIERE: Telecommunication

Networking

Theme

Automatic Arabic Speech Recognition by CNN

Thesis Jury:

Name and Surname	Grade	Quality
Dr. Chellali Safouane	MCA	Supervisor
Dr. Birane Mouhoub	MAA	President

Promotion: 2022/2023


بسم الله الرحمن الرحيم

Acknowledgment

At the end of this Honest work, we extend our thanks to all

the persons who have contributed directly or indirectly so that we

may complete this completion project.

We deeply thank Mr. M.KORIBA and Mr. M.REGUIRGUE for their quality supervision, the efforts they made in trying to perfect our work, for their valuable help and advice and sympathy.

Our thanks also go to Mr.Bouzaida for his leadership in the Transmition Centre and for all the information he has made available to us.

Our thanks also go to the members of the jury for agreeing

to examine our work.

Special thanks to all college workers and professors.

I extend my profound appreciation to: Dr.Merrah Lahcen , Dr.Saadi Ramdani, Dr.Chaker Saleh, Dr. seghir Abdelkader, Dr.Zitouni Abdelkader And Dr.Regab Ilyes, Dr.Ourdess Asma.

And To Dr.Tedjani Nassima . رحمة الله عليها

Finally, our thanking our families and especially our parents for us to have supported and to have done everything so that we can achieve an advanced stage of success. We will never be grateful enough to them.

Dedication

As I stand at the culmination of this arduous journey, the completion of my thesis, my heart overflows with a profound sense of gratitude and emotion. Today, I dedicate this thesis to my parents, my unwavering source of inspiration and the beating pulse behind my every endeavor.

Throughout this demanding undertaking, you have been my unwavering rock, my pillar of strength, and my eternal wellspring of encouragement. Your unwavering belief in my abilities and relentless support have propelled me forward when doubt threatened to engulf my spirit. Your faith in me, even during moments of self-doubt, has reminded me of the boundless potential that lies within.

I extend my gratitude appreciation to my beloved brothers Badis, Lotfi and my lovely sister Batoul for being the best company in my life.

Furthermore, I extend my sincerest gratitude to my colleague and cherished friend, Mohamed Bachir Khelifi, whose unwavering support and camaraderie have been invaluable throughout this expedition.

Additionally, I extend my profound appreciation to brethren who have accompanied me on this journey Lahcen, abdou, Mouhamed, houcine, Khaled.

lastly, I am overwhelmed with an outpouring of deep gratitude towards those who have graced my path, regardless of their proximity or the mere whisper of a word. Your unwavering support and invaluable aid have etched an indelible impression upon the depths of my soul.

May this dedication serve as a tangible token of my gratitude, a humble reflection of the profound impact you have had on my life and my pursuit of knowledge. As I step forward into the next phase of my journey, I carry with me the indelible mark you have left upon my heart and soul.

SEFARI Ali Elhocine

Dedication

With utmost admiration and heartfelt gratitude, I humbly dedicate this dissertation to an extraordinary man whose physical presence eluded me, yet his essence resided eternally within the depths of my soul and the chambers of my heart. To the individual whom I deeply revered, whose indelible legacy resonates within me, my beloved father, KHELIFI THAMER, may you find solace in a realm of boundless serenity.

Additionally, I extend my profound appreciation to the remarkable woman who tirelessly assumed the roles of both nurturer and protector, courageously navigating the tumultuous currents of the world to ensure our unity. To GUERMIT KHEIRA, whose unwavering support fortified my spirit, I offer my sincerest gratitude.

Moreover, I extend my unwavering affection and profound admiration to the love of my life, my confidante, and my closest companion. To my cherished sister, KHELIFI ZAHRA, who consistently stood as my unwavering advocate, I express immeasurable gratitude.

To my dear brethren who have accompanied me on this journey, Saad, Mohammed, Aziz, and Abdeslam, I am indebted to you for the unwavering encouragement and infectious mirth that illuminated my path during this phase of my existence.

In reverent recollection, I acknowledge the indomitable spirit of KHADIDJA and the entire SRH classroom, as I pledge to cherish the profoundly captivating moments we shared together.

Furthermore, I extend my sincerest gratitude to my colleague and cherished friend, ALI SEFARI, whose unwavering support and camaraderie have been invaluable throughout this expedition.

Lastly, I express profound gratitude to all those who have extended their assistance, whether from near or far, even though a single word. Your contributions have left an indelible mark upon my heart. To each and every one of you, I offer my heartfelt appreciation.

In conclusion, I convey my deep affection to all, as your presence has enriched my journey immeasurably

KHELIFI Mohammed Elbachi

Summary

List of figures

List of tables:

ABBREVIATION

ASR: Automatic Speech Recognition.

FFT: Fast Fourier Transform.

IN: Intelligent Network.

LPC: Linear Prediction Coefficients.

LPCC: Linear prediction cepstral coefficients.

MFCC: Mel-Frequency Cepstral Coefficients.

PLP: Perceptual Linear Prediction.

SAMPA: Speech Assessment Methods Phonetic Alphabet.

PER: phoneme error rate.

WER: word error rate.

CNN: Convolutional Neural Networks

General Introduction

General Introduction:

In recent years, significant progress has been made in the field of automatic speech recognition (ASR) due to advancements in machine learning and deep learning algorithms. This progress has also extended to the domain of Arabic speech recognition, considering Arabic's widespread usage and unique phonetic characteristics. The main objective of this thesis is to explore the effectiveness of Convolutional Neural Networks (CNNs) in Arabic speech recognition and demonstrate their capabilities through accurate examples in different domains.

However, it is essential to acknowledge the challenges that arise in Arabic speech recognition. These challenges stem from the phonetics, phonology, and dialectal variations present in the Arabic language. The complexity of Arabic script and pronunciation, along with limited labeled data for training purposes, further compound the difficulties. Additionally, regional accents add to the complexity, making accurate recognition a demanding task.

To overcome these challenges, the thesis delves into the process of preparing Arabic speech datasets, specifically addressing issues related to dialectal variations and noise. It explores the design of model architectures, hyperparameter tuning, and regularization techniques specifically tailored for Arabic speech recognition using CNNs. The evaluation of the developed models employs metrics such as accuracy, word error rate (WER), and phoneme error rate (PER), providing a means to compare CNN-based ASR models with traditional approaches.

Chapiter I:
Speech signal analysis methods

Chapiter I: Speech signal analysis methods

Introduction:

Speech recognition, a pivotal technology in the field of human-computer interaction, has gained immense significance in modern technology due to its ability to bridge the gap between human communication and machines. This paper presents an academic introduction that emphasizes the importance of speech recognition in various applications and highlights its transformative impact on modern technology. It explores the advantages and challenges of speech recognition, along with its potential for revolutionizing diverse domains.

This section focuses on how speech recognition has revolutionized accessibility for individuals with disabilities. It discusses the empowerment brought by voice-controlled interfaces, enabling people with mobility or visual impairments to interact with technology in a natural and efficient manner.

1. CLASSIFICATION OF SPEECH RECOGNITION SYSTEMS:

Speech recognition systems can be separated in several different classes by describing the type of speech utterance, type of speaker model, type of channel and the type of vocabulary that they have the ability to recognize. Speech recognition is becoming more complex and a challenging task because of this variability in the signal. These challenges are briefly explained below.

1.1.Types of Speech Utterance:

An utterance is the vocalization (speaking) of a word or words that represent a single meaning to the computer. Utterances can be a single word, a few words, a sentence, or even multiple sentences. The types of speech utterance are:

1.1.1. Isolated Words:

Isolated word recognizers usually require each utterance to have quiet on both sides of the sample window. It doesn't mean that it accepts single words, but does require a single utterance at a time.

1.1.2. Connected Words:

Connected word systems, or more accurately "connected utterances," are similar to isolated words in that they enable separate utterances to be "run-together" with minimal space between them.

1.1.3. Continuous Speech:

Continuous speech recognizers allow users to speak almost naturally, while the computer determines the content. Basically, it's computer dictation.

1.1.4. Spontaneous Speech:

This type of speech is natural and not rehearsed. An ASR system with spontaneous speech should be able to handle a variety of natural speech features such as words being run together and even slight stutters.

1.2.Types of Speaker Model:

1.2.1. Speaker dependent models:

Speaker dependent systems are designed for a specific speaker. They are generally more accurate for the particular speaker, but much less accurate for other speakers.

1.2.2. Speaker independent models:

Speaker independent systems are designed for variety of speakers. It recognizes the speech patterns of a large group of people.

2. Speech signal analysis method

Speech is an auditory signal that carries the ideas formulated in the speaker's mind. It not only conveys language information but also provides insights into various aspects such as sounds, syntax, and the speaker's characteristics, including age, gender, local origin, health, emotional state (mood), and unique traits. Automatic Speech Recognition (ASR) solely focuses on the acoustic information embedded in the speech signal, necessitating the use of acoustic analysis.

The purpose of acoustic analysis is to derive representative coefficients from the speech signal. These coefficients are computed at regular intervals of time. In essence, the speech signal is transformed into a sequence of coefficient vectors that must accurately represent the intended model and capture a maximum amount of useful information for recognition.

Before calculating the parameters, this analysis requires preprocessing of the speech signal.

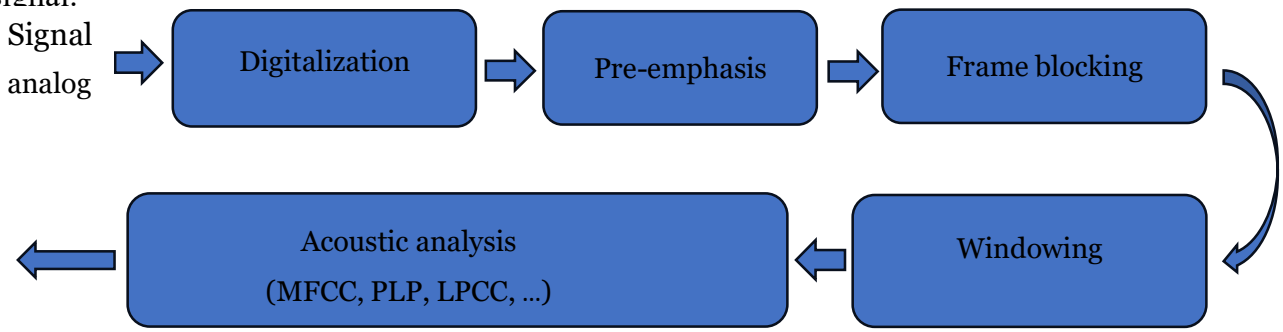

Figure 1. Pre-processing of the speech signal

2.1.Pre-processing of the speech signal

2.1.1. Digitization

Signals we use in the real world, such as our voice, are analog signals continuous in both time and amplitude to process these signals for digital communication, we need to convert it to "digital" form (discrete in both time and amplitude) for this, we use Nyquist–Shannon sampling theorem, the sampling frequency fs is greater than or equal to the twice the highest frequency component of the message signal:

$F_s \geq 2F_{max}$ (I.1)

F_{max} is the maximum frequency component of the analog signal to be sampled.

2.1.2. Pre-emphasis:

Pre-emphasis is a technique used in the way of speech processing to enhance high frequenciesof the signal, so it reduces the high spectral dynamic range.

We do that by passing the signal through a high-pass filter that has the following transfer function:

$H(z) = 1 - az^{-1}$ (I.2)

with: $0.9 < a < 0.98$

With that we insure that we spectrally flattened the signal and made it less susceptible to finite precision effects.

2.1.3. Frame blocking:

The statistical features of a speech signal are generally invariant within a short time interval with frame blocking method the pre-emphasized signal is blocked, segmented into frames with a common frame length spaced between 20-25ms apart.

2.1.4. Windowing:

When we complete frame blocking procedure, we apply a windowing function to every frame we took to eliminate the effect of discontinuities at frame edges.

The procedure of windowing is done by multiplying every frame we took to the window Hamming w[n] with length N+1 which is given by:

$$\mathrm{W[n]} = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{Ah}\right) & ,0 \le n \le N \\ 0 & \text{, otherwise} \end{cases} \qquad \text{(I.3)}$$

Where N is the length of frame.

2.2.Fourier transform analysis:

Fast Fourier Transform (FFT) is the traditional technique to analyze frequency spectrum of the signal in speech recognition.

FFT produce frequency spectrum which contains all the information about original signal, but in different form. It is frequency domain representation of signal.

The Fast Fourier Transform is given in the following equation:

$$\mathrm{X}(k) = \sum_{\mathrm{j}=1}^{N} \frac{-2\pi \mathrm{i}}{N^m}(\mathrm{j}-1)(k-1) \; x(\mathrm{j})e \qquad \text{(I.4)}$$

The FFT takes advantage of the symmetry and periodicity properties of the Fourier Transform to reduce computation time. In this process, the transform is partitioned intoa sequence of reduced-length transforms that is collectively performed with reduced computation [1].

2.3.Linear Predictive Coding (LPC):

Linear Predictive Coding (LPC) is a widely employed method for speech coding and representation, extensively utilized in the field of speech recognition. LPC analysis serves as a prominent technique, facilitating the representation of a signal through a concise set of parameters derived via straightforward computations.

The basic idea of LPC method is that a given speech sample s(n) at time n can be represented as a linear combination of p previous speech samples weighted with some coefficients ak, such that [2]:

$$S(n) = a_1 s(n-1) + a_2 s(n-2) + \cdots + a_p(n-p) + G(n) \qquad \text{(I.5)}$$

where the coefficients a1, a2, ..., ap are assumed constant over the speech analysis frame We convert Equation (II. 5) to an equality by including year-excitation term Gu(n), giving:

$$s(n) = \sum_{k=1}^{Z} a_k s(n-k) + Gu(n) \qquad \text{(I.6)}$$

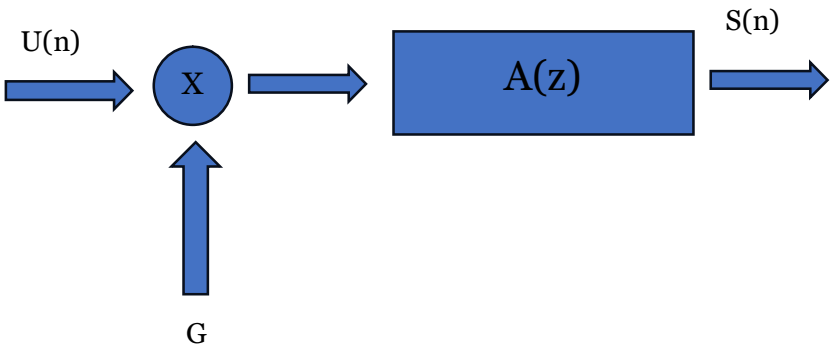

Figure 2. Linear prediction model of speech

Based on the model of figure (I.2), the exact relationship between s(n) and u(n) is:

$$s(n) = \sum_{k=1}^{a_k s(n-k)+Gu(n)} \qquad \text{(I.7)}$$

We consider the linear combination of past speech sample as the estimate, defined as:

$$s'(n) = \sum_{k=1} a_{ks}(n-k) \qquad \text{(I.8)}$$

We now from the predictive error, e(n), defined as:

$$e(n) = s(n) - s'(n) = s(n) - \sum_{k=1} a_k s(n-k) \qquad \text{(I.9)}$$

To minimize (E) by choice of the coefficients a_k, differentiate with respect to each of them and set the resulting derivatives to zero:

$$\frac{\partial E_v}{\partial a_{\mathrm{i}}} = 0 \quad (\mathrm{i} = 1,2,\dots\dots p) \qquad \text{(I.10)}$$

the calculation of this relationship leads to the following equation:

$$\sum_{k=1}^{p} a_k \phi_{ik} = -\Phi_{i0} \qquad i = 1,2,\dots\dots,p \qquad \text{(I.11)}$$

$$\Phi_{ik} = \sum_{n=n1}^{n^2} s(n-\mathrm{i})s(n-k) \qquad \text{(I.12)}$$

This normal equation (I.11), called Yule-Walker, constitute a linear system of P equations with P unknowns the resolution of this system will make it possible to obtain the coefficients of the filter

Among the methods of minimization of the residual energy of prediction thus of resolution of the system, we find mainly the autocorrelation method and the covariance method

2.3.1. the auto correlation method:

Because of the time-varying nature of the speech signal, the predictor coefficients must be estimated from short segments of speech signals (10-40ms) where the characteristics of speech signals are constant in this range [3]

The residual energy of prediction minimized in ±∞:

$$E_p = \sum_{n=-\infty}^{+\infty} e^2(n) \qquad \text{(I.13)}$$

Knowing that samples outside the prediction error interval are all zero, this is equal tomultiplying the signal by a window of finite length corresponding to N samples

If s(n) is nonzero only for $0 \leq n \leq N-1$, then the corresponding prediction error,e(n) for an order predictor will be nonzero over the interval $0 \leq n \leq N-1+p$

Thus, for this case E_p is properly expressed as:

$$E_p = \sum_{n=0} \sum_{n-1+p} e^2(n) \qquad \text{(I.14)}$$

And $\phi_{(ik)}$ Can be expressed as:

$$\Phi_{iK} = \sum_{n=0}^{N-1+p} s(n-\mathrm{i})s(n-k) \text{ with: } 1 \leq \mathrm{i} \leq p_k 0 \leq k \leq p \qquad \text{(I.15)}$$

Gold:

$$\Phi_{\mathrm{i}k} = \sum_{n=0}^{N-1-(\mathrm{i}-k)} s(n)s(n+\mathrm{i}-k) \qquad \text{(I.16)}$$

Furthermore it can be seen that in this case ϕ_{ik} is identical to the short time autocorrelation function, where:

$$R_k = \sum_{n=0}^{N-1-k} s(n)s(n+k) \qquad \text{(I.17)}$$

Since R_k is an even function it follows that:

$$\phi_{ik} = R(|i-k|) \qquad \text{(I.18)}$$

Therefore eq. (I.12) can be expressed as:

$$R_i = \sum_{k=1}^{p} a_k R(|i-k|) \qquad \text{with: } 1 \leq i \leq p \qquad \text{(I.19)}$$

The set of equation (I.19) can be expressed in matrix form of (p * p) autocorrelation value is a Toeplitz matrix, i.e., it is symmetric and all the elements along a given diagonal are equal. Several efficient recursive procedures have been devised for solving this system of equations; the most efficient method known for solving this particular system of equations is Durbin's recursive procedure [4].

2.3.2. The covariance method:

The signal is extended by p samples outside the normal range of $0 \leq n \leq N-1$ to include p samples occurring prior to n=0 (they are available) and eliminates the need fora tapering window

$$E_p = \sum_{n=0}^{N-1} e^2(n) \qquad \text{(I.20)}$$

ϕ_{ik} Can be expressed as:

$$\Phi_{ik} = \sum_{n=0}^{N-1} s(n-\mathrm{i})s(n-k) \qquad \text{(I.21)}$$

Or by changing the variable:

$$\Phi_{\mathrm{i}k} = \sum_{n=-\mathrm{i}}^{N-\mathrm{i}-1} s(n)s(n+\mathrm{i}-k) \qquad \text{(I.22)}$$

Using the extended speech interval to define the covariance values ϕ_{ik} , the resulting covariance matrix is symmetric but not Toeplitz, and can be solved efficiently by a set of techniques called the Cholesky decomposition method [5].

2.4.Cepstral analysis:

Speech signals are comprised of two fundamental components: the excitation source and the vocal tract system. The generation of speech can be conceptualized as the convolution of the excitation sequence and the characteristics of the vocal tract filter. Let us denote the excitation sequence as e(n) and the vocal tract filter sequence as h(n). Accordingly, the speech sequence, denoted as s(n), can be mathematically expressed as follows:

$$s(n) = e(n) * h(n) \qquad \text{(I.23)}$$

In order to achieve a comprehensive analysis and modeling of the excitation and system components within speech, as well as their subsequent utilization in diverse speech processing applications, it is imperative to effectively separate these two components from the speech signal.

The primary aim of cepstral analysis is to achieve the disentanglement of the excitation source and vocal tract filter, denoted as h(n), in the absence of any prior knowledge regarding the source and/or vocal tract system. This is accomplished through the application of a homomorphic transformation, which replaces the convolution operation with a summation. The cepstrum, a key outcome of cepstral analysis, is

obtained by performing an inverse Fourier transform (IFT) on the logarithm of the estimated spectrum of a signal.

2.4.1. LPC Parameter Conversion to Cepstral Coefficients:

Cepstral analysis is widely utilized in the Domain of speech processing due to its inherent capability to accurately represent speech waveforms and their characteristics through a compact set of features.

Linear prediction cepstral coefficients (LPCC) are cepstral coefficients derived from LPC calculated spectral envelope. They are the coefficients of the Fourier transform illustration of the logarithmic magnitude spectrum of LPC.

In speech processing, LPCC analogous to LPC, are computed from sample points of a speech waveform, the horizontal axis is the time axis, while the vertical axis is the amplitude axis. LPCC (C_m) can be calculated using:

$$C_0 = \ln G^2 \qquad \text{(I.24)}$$

Where, G^2 is the gain term in LPC model.

$$C_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) C_k a_{m-k} \text{ with: } 1 \leq m \leq p \qquad (I.25)$$

$$C_m = \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) C_k a_{m-k} \text{ with: } m > p \qquad (I.26)$$

Where, a_m is the linear prediction coefficient, C_m is the cepstral coefficient.

2.4.2. Mel-Frequency Cepstral Coefficients (MFCC):

Mel-Frequency Cepstral Coefficients (MFCC) is a widely used feature extraction technique in the field of speech and audio signal processing. It plays a crucial role in various applications such as speech recognition, speaker identification, and music genre classification. This essay aims to provide an accurate overview of MFCC, explaining its underlying principles, steps involved in the computation, and its significance in signal processing applications.

Mel Frequency Cepstral Coefficients (MFCCs) have emerged as a widely adopted method for extracting distinctive features in the realm of speech recognition. Leveraging insights from the non-linear characteristics of the human auditory perception system, MFCCs utilize the concept of the 'Mel Scale' to capture the subjective pitch associated with each tone [6]. The relationship between the actual frequency f measured in Hertz (Hz) and its corresponding pitch in mels, denoted as B(f), can be defined by the following mathematical relation:

$$B(f) = 2595\log_{10}\left(1 + \frac{f}{700}\right) \qquad \text{(I.27)}$$

Let a discrete signal {x [n]} with 0 ≤ n ≤ N-1, N is the number of samples of a window, analyzed, Fs is the sampling frequency, the discrete Fourier transform S [k] is obtained:

$$S[k] = \sum_{n=0}^{N-1} x[n]e^{-\frac{i2\pi nk}{N}} \text{ with: } 0 \leq k \leq N \qquad \text{(I.28)}$$

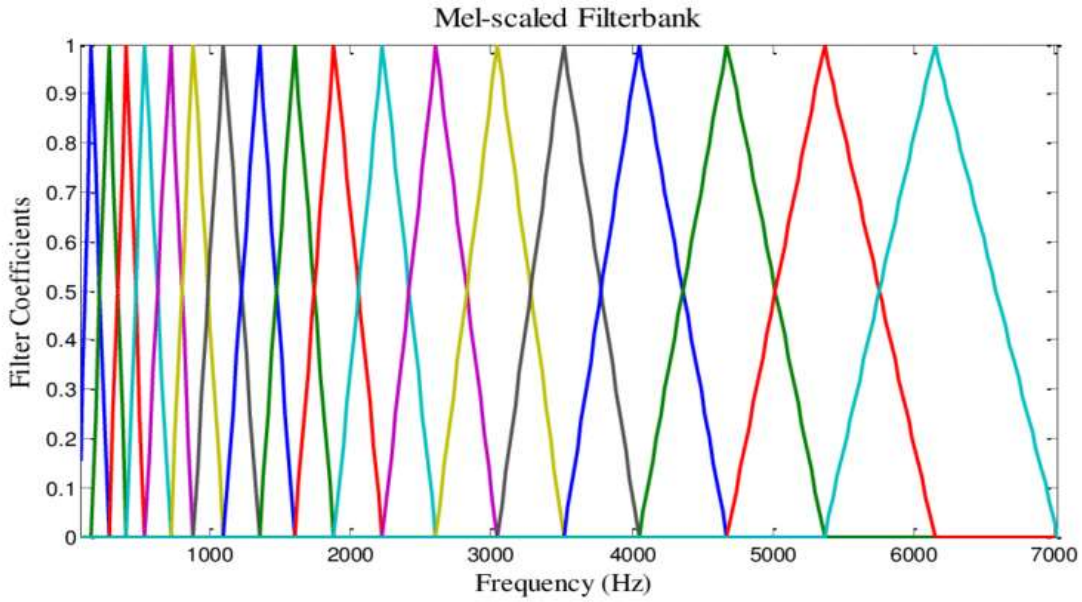

Figure 3. Mel filter bank

The spectral characteristics of the signal are subjected to multiplication with triangular filters, as depicted in Figure 3. These filters possess equivalent bandwidths within the Mel-frequency domain. The determination of the boundary points B[m] for the frequency filters is carried out according to the following calculation method:

$$B[m] = B(f_1) + m\frac{B(f_h)-B(f_1)}{M+1} \qquad \text{with: } 0 \le m \le M+1 \qquad \text{(I.29)}$$

where M is the number of filters, f h is the highest frequency and f 1 is the lowest frequency for signal processing. In the frequency domain, the corresponding discrete f [m] points are calculated by the equation:

$$f[m] = \left(\frac{N}{F_s}\right) B^{-1}\left(B(f_1) + m\frac{B(f_h)-B(f_1)}{M+1}\right) \qquad \text{(I.30)}$$

where B^{-1} is the Mel-frequency frequency transform:

$$B^{-1}(b) = 700 * \left(10^{\frac{b}{2595}} - 1\right) \qquad \text{(I.31)}$$

The coefficient Hm [k] of each filter is determined by the following system:

$$H_m[k] = \begin{cases} 0 & \text{if } k \le f[m-1] \\ \frac{k - f[m-1]}{f[m] - f[m-1]} & \text{if } f[m-1] \le k \le f[m] \\ \frac{f[m+1] - k}{f[m+1] - f[m]} & \text{if } f[m] \le k \le f[m-1] \\ 0 & \text{if } k \ge f[m+1] \end{cases} \quad \text{(I.32)}$$

To insure a smooth and stable spectrum, at the output of the filters a logarithm of energy (or a logarithm of amplitude spectrum) is calculated:

$$E[m] = \log\left[\sum_{k=0}^{N-1} \mid S[k]^2 H_m[k]\right] \text{ with } 0 \le \text{m} \le \text{M} \quad \text{(I.33)}$$

The cepstral coefficients of Mel-frequency (MFCCs) can be obtained by a discretecosine transform of E [m]:

$$C[n] = \sum_{m=0}^{M-1} E[m] \cos\left(\frac{pn\left(m+\frac{1}{2}\right)}{M_S}\right) \text{ with } : 0 \le n < M \quad \text{(I.34}$$

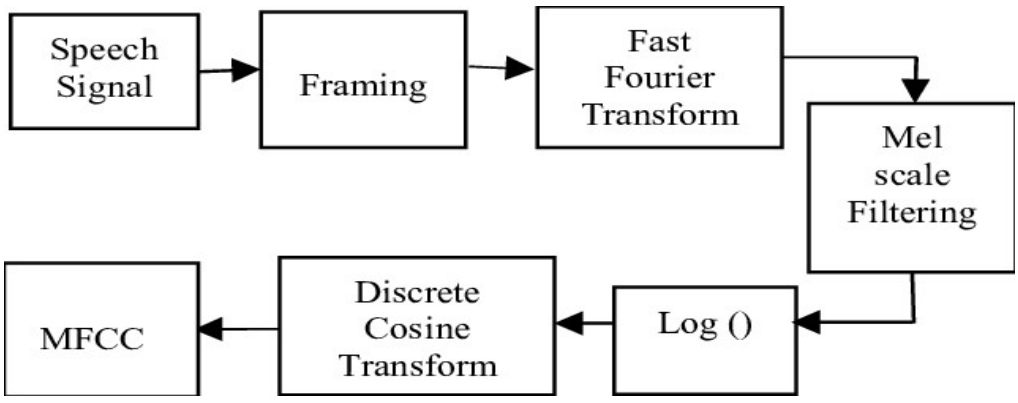

Figure 4. Block Diagram of MFCC Processing system

Mel Frequency Cepstral Coefficients (MFCC) represent an audio feature extraction technique that effectively captures speech-related parameters akin to those employed by the human auditory system. This technique selectively emphasizes the relevant speech

characteristics while simultaneously attenuating extraneous information present in the signal.

2.4.3. Perceptual Linear Prediction (PLP):

The Perceptual Linear Prediction (PLP) model, formulated by Herman Sky, constitutes an advanced approach in the field of telecommunications, specifically in the realm of speech analysis and recognition. Inspired by the principles of psychophysics of hearing [6, 7], PLP aims to replicate the perceptual characteristics of human speech perception. By selectively discarding extraneous information, the PLP model enhances the accuracy of speech recognition. Although similar to Linear Prediction Coding (LPC) in its core methodology, PLP incorporates spectral transformations to align the spectral characteristics of the speech signal with those of the human auditory system.

PLP approximates three main perceptual aspects namely: the critical-band resolution curves, the equal-loudness curve, and the intensity-loudness power-law relation, which are known as the cubic-root. Figure II.5 shows steps of PLP computation [8].

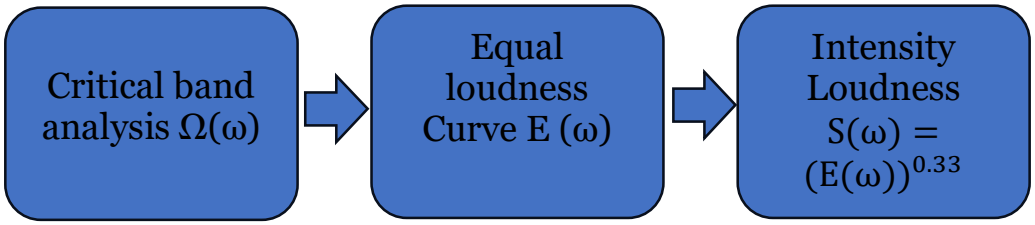

Figure 5. Steps of plp computation

After shaping the speech signal, the power spectrum P (w) is calculated. Then, a passage of the usual frequency scale on the Bark scale is performed.

$$\Omega(w) = 6\ln\left(\frac{w}{1200} + \left(\left(\frac{w}{1200p}\right)^2 + 1\right)^{\frac{1}{2}}\right) \qquad (I.35)$$

W is representing the angular frequency expressed in rd / s and Ω the frequency of Bark.

This passage to the Bark scale makes it possible to roughly approximate what we know about the shape of the hearing filters. It is approximately constant along the Bark scale. The power spectrum in the Bark scale is convoluted with the power spectrum of the critical band curve using the following equation:

$$\Psi(\Omega) = \begin{cases} 0 & \text{if } \Omega \leq -1.3 \\ 10^{2.5+(\Omega+0.5)} & \text{if } -1.3 \leq \Omega \leq -0.5 \\ 1 & \text{if } -0.5 \leq \Omega \leq 0.5 \\ 10^{-1.0(\Omega-0.5)} & \text{if } 0.5 \leq \Omega \leq 2.5 \\ 0 & \text{if } \Omega \geq 2.5 \end{cases} \qquad \text{(I.36)}$$

This Masking curve is an approximation of asymmetric Schroeder masking curvature.

Then, using a transfer function E (w), we attempt to approximation the sensitivity of the human ear at various frequencies. This transfer function multiplies the power spectrum.

$$E(\Omega) = E(w) + (\Omega) \qquad \text{(I.37)}$$

$$\theta(\Omega_t) = \sum_{\Omega=-1.3}^{\Omega=2.3} p(\Omega - \Omega_t)\psi(\Omega) \qquad \text{(I.38)}$$

A power loss law is then used to approximate the non-linearity between the intensity of a sound and its force of perception by the ear:

$$\phi(\Omega) = E(\Omega)^{0.33} \qquad \text{(I.39)}$$

The next stage is to perform a traditional autoregressive modeling of the all-pole auditory model, determining the filter's autoregressive coefficients.

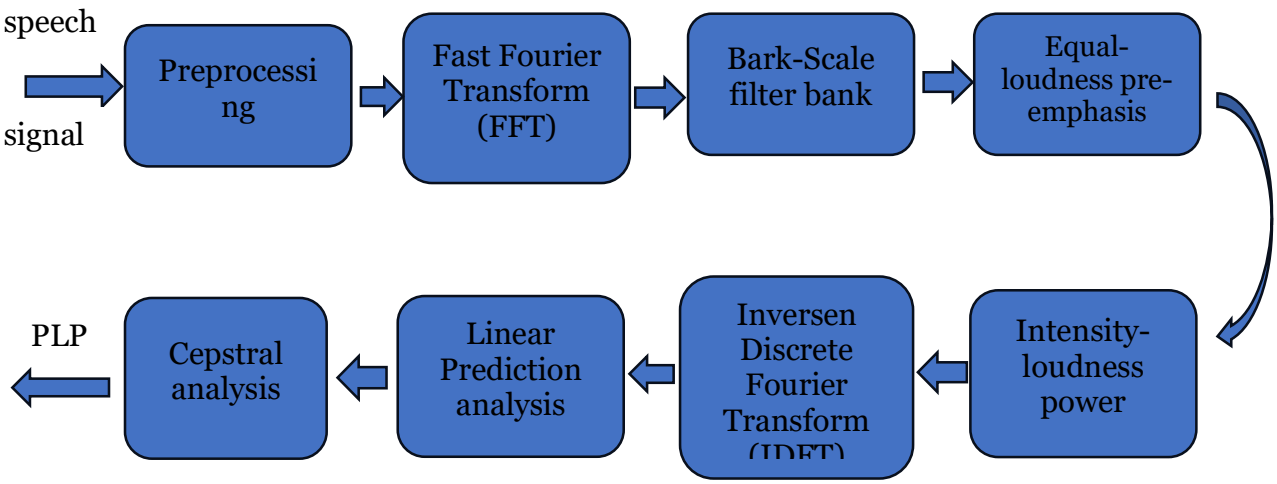

Figure 6. Block Diagram of PLP Processing

Conclusion

In this chapter, we discussed the most essential analysis approaches for extracting the important properties of a speech signal, such as LPC (linear predictive coefficients), MFCC (Mel-Frequency Cepstral Coefficients), and PLP (Perceptual Linear Prediction).

Chapter II:
An Introduction to CNNs and Deep Learning

Chapter II: An Introduction to CNNs and Deep Learning

Introduction:

A Convolutional Neural Network (CNN) is a Deep Learning neural network design that is widely utilized in computer vision. Computer vision is an artificial intelligence field which allows a computer to comprehend and analyze images or visual data. And CNN is a deep learning model for processing data with a grid pattern, such as photographs, that is inspired by the organization of animal visual cortex and meant to learn spatial hierarchies of characteristics automatically and adaptively, from low- to high-level patterns. CNN is a mathematical architecture made up of three types of layers (or building blocks): convolutional, pooling, and fully connected.

The first two layers, convolution and pooling, extract features, while the third, a fully connected layer, transfers the extracted features into final output, such as classification. A convolution layer is essential in CNN, which is made up of a stack of mathematical operations such as convolution, a sort of linear operation. Pixel values in digital images are stored in a two-dimensional (2D) grid, i.e., an array of numbers, and a small grid of parameters called kernel, an optimizable feature extractor, is applied at each image position, making CNNs extremely efficient for image processing because a feature can occur anywhere in the image. Extracted features can become hierarchically and progressively more complex as one layer feeds its output into the next layer. Training is the process of optimizing parameters such as kernels in order to minimize the difference between outputs and ground truth labels using optimization algorithms such as backpropagation and gradient descent, among others.

Artificial Neural Networks perform exceptionally well in Machine Learning. Neural Networks are employed in a wide range of datasets, including images, audio, and text. For example, to predict the sequence of words, we use Recurrent Neural Networks, more precisely an LSTM, and for picture classification, we use Convolutional Neural Networks. In this blog, we will construct a fundamental building component for CNN.

CNNs are used in so many applications now:

- Object recognition in images and videos (think image-search in Google, tagging friends faces in Facebook, adding filters in Snapchat and tracking movement in Kinect)
- Natural language processing (speech recognition in Google Assistant or Amazon's Alexa)
- Playing games (the recent defeat of the world 'Go' champion by DeepMind at Google)
- Medical innovation (from drug discovery to prediction of disease)

CNN or Deep Learning?

The "deep" element of deep learning manifests itself in two ways: the number of layers and the number of features. To begin, as one might anticipate, a deep learning framework often has more layers than a multi-layer perceptron or regular neural network. We have some architectures with 150 layers. Second, each layer of a CNN will learn numerous 'features' (many sets of weights) that connect it to the preceding layer; hence, it is significantly deeper than a standard neural net in this sense as well. In fact, some of the most powerful neural networks, like CNNs, include only a few layers. As a result, the 'deep' in DL recognizes that each layer of the network learns many features.

Although CNNs and DL are frequently used interchangeably, the concept of DL predates CNNs by some time. Connecting several neural networks, changing the directionality of their weights, and stacking such machines all contributed to DL's growing power and appeal.

As with neural network research, the idea for CNNs originated from nature, notably the visual cortex. It was based on the premise that neurons in the visual cortex focus on different sized portions of a picture, receiving varying degrees of information in different layers. If a computer could be made to work in this manner, it might be able to imitate the brain's image-recognition power. So, how can this be accomplished?

A CNN takes an array or an image (2D or 3D, grayscale or color) as input and attempts to learn the link between this image and some target data, such as a classification. We're still talking about weights when we say 'learn', just like in a standard neural network. The distinction in CNNs is that these weights connect short segments of the input to each of the neurons in the first layer. In essence, several neurons in a single layer each have their own weights to the same portion of the input. These various weight sets are referred to as "kernels." [9].

1. CNN architecture:

An overview of the architecture and training process of a convolutional neural network (CNN). A CNN is composed of several building blocks, including convolution layers, pooling layers (e.g., max pooling), and fully connected (FC) layers. The performance of a model under specific kernels and weights is calculated with a loss function via forward propagation on a training dataset, and learnable parameters, i.e., kernels and weights, are updated according to the loss value via backpropagation using the gradient descent optimization algorithm. ReLU stands for rectified linear unit.

The input layer, Convolutional layer, Pooling layer, and fully connected layers make up a Convolutional Neural Network. The Convolutional layer extracts feature from the input image using filters, the Pooling layer reduces computation by down sampling

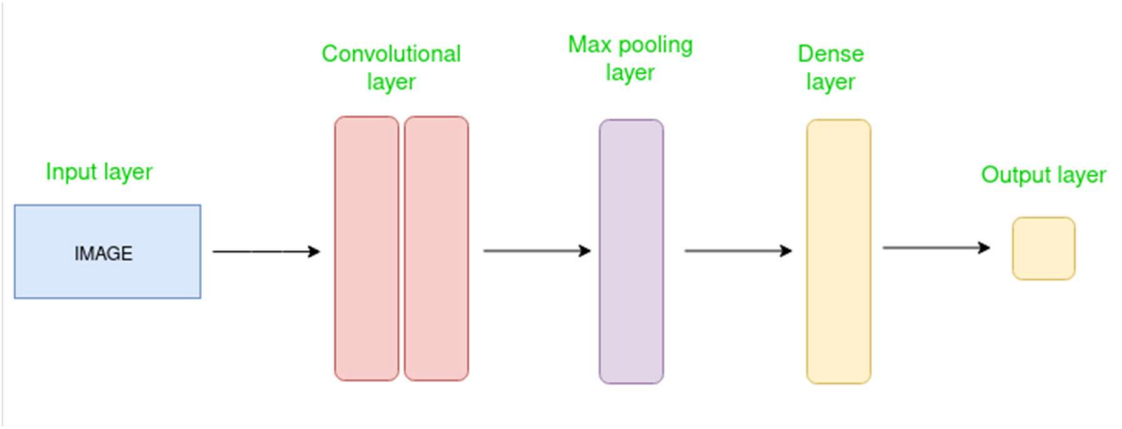

Figure 7. CNN simple architecture

the image, and the fully

Connected layer makes the final prediction. Backpropagation and gradient descent are used by the network to learn the best filters.

2. How Convolutional Layers works?

Convolutional Neural Networks, often known as covnets, are neural networks that share parameters. Assume you have an image. It can be represented as a cuboid with length, width (picture size), and height (i.e. the channel, as images often have red, green, and blue channels).

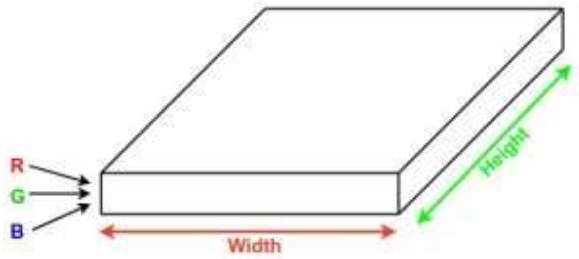

Figure 8. Explaining image

Consider taking a small section of this image and running a little neural network, known as a filter or kernel, on it with, say, K outputs and representing them vertically. Slide that neural network across the entire image, and we'll obtain a new image with varying widths, heights, and depths. Instead of simply the R, G, and B channels, we now have more channels but with a smaller width and height. Convolution is the name given to this procedure. If the patch size is the same as the image size, the neural network is a normal neural network. We have fewer weights as a result of this tiny patch.

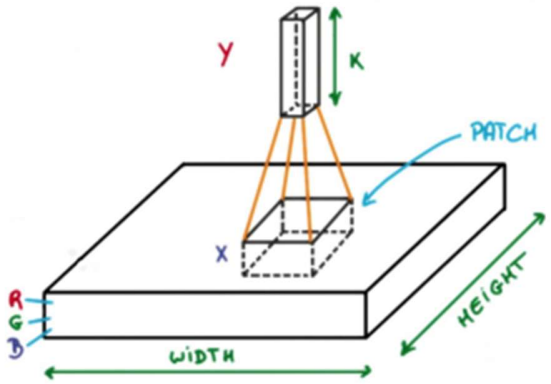

Figure 9. Explaining image

Now let's talk about a bit of mathematics that is involved in the whole convolution process.

- Convolution layers consist of a set of learnable filters (or kernels) having small widths and heights and the same depth as that of input volume (3 if the input layer is image input).
- For example, if we have to run convolution on an image with dimensions 34x34x3. The possible size of filters can be axax3, where 'a' can be anything like 3, 5, or 7 but smaller as compared to the image dimension.
- During the forward pass, we slide each filter across the whole input volume step by step where each step is called stride (which can have a value of 2, 3, or even 4 for high-dimensional images) and compute the dot product between the kernel weights and patch from input volume.
- As we slide our filters, we'll get a 2-D output for each filter and we'll stack them together as a result, we'll get output volume having a depth equal to the number of filters. The network will learn all the filters [10].

3. Layers used to build ConvNets:

The term covnets refers to a complete Convolutional Neural Networks design. A covnet is a series of layers, each of which changes one volume to another using a differentiable function.

3.1.Input Layers:

The input image is placed into this layer. It can be a single-layer 2D image (grayscale), 2D 3-channel image (RGB color) or 3D. The main difference between how the inputs are arranged comes in the formation of the expected kernel shapes. Kernels need to be learned that are the same depth as the input i.e. 5 x 5 x 3 for a 2D RGB image with dimensions of 5 x 5. Inputs to a CNN seem to work best when they're of certain dimensions. This is because of the behavior of the convolution. Depending on the *stride* of the kernel and the subsequent *pooling layers* the outputs may become an

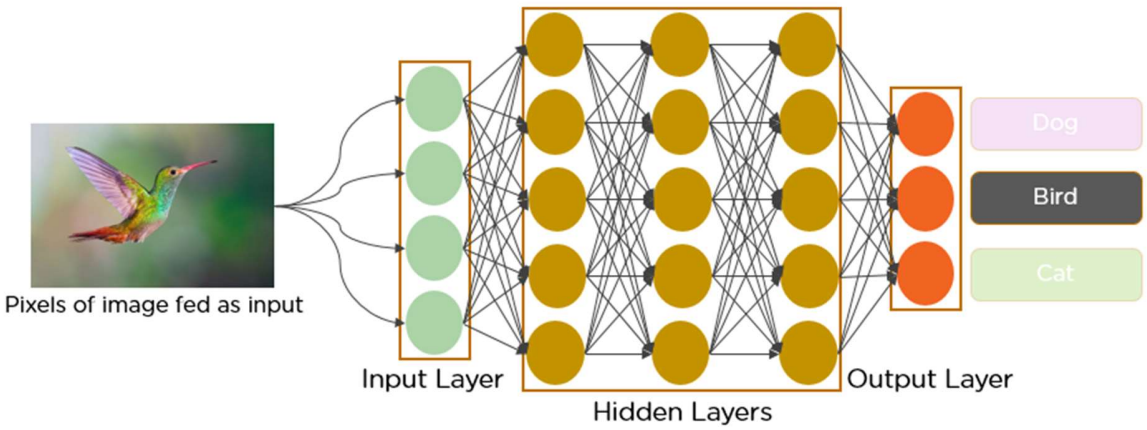

Figure 10. Explaining figure of input layers

"illegal" size including half-pixels. We'll look at this in the pooling layer section.

3.2.Convolutional Layers:

This is the layer, which is used to extract the feature from the input dataset. It applies a set of learnable filters known as the kernels to the input images. The filters/kernels are

smaller matrices usually 2×2, 3×3, or 5×5 shape. it slides over the input image data and computes the dot product between kernel weight and the corresponding input image patch. The output of this layer is referred ad feature maps [11].

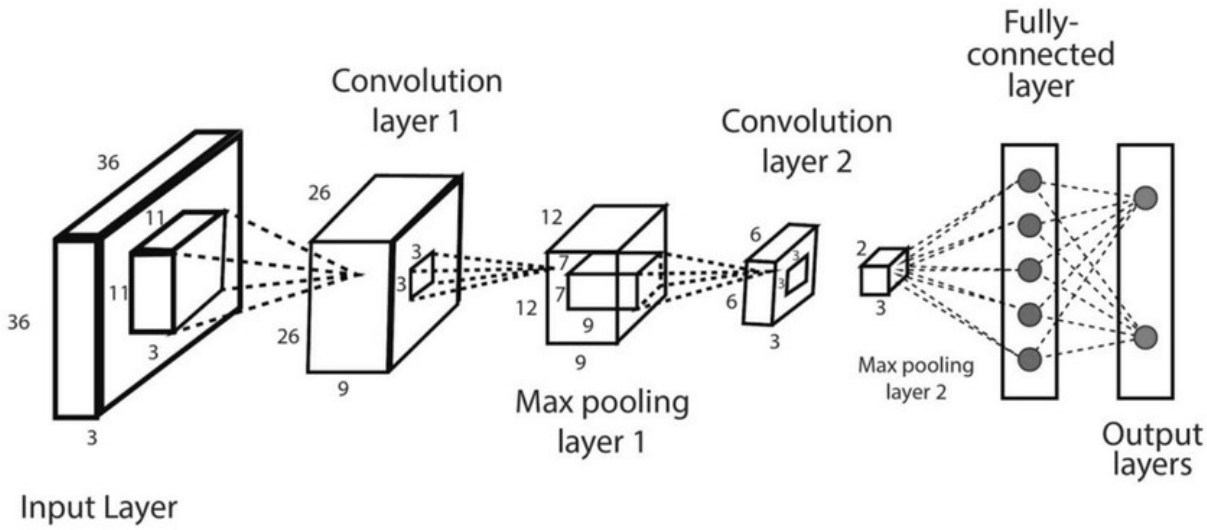

Figure 11. Explaining figure of convolutional layers

3.3.Activation Layer:

By adding an activation function to the output of the preceding layer, activation layers add nonlinearity to the network. it will apply an element-wise activation function to the

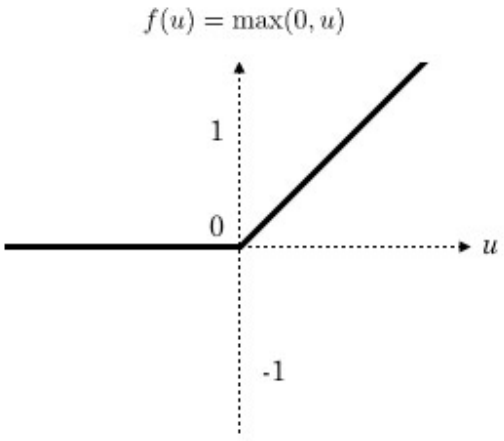

Figure 12. RELU function

output of the convolution layer. Some common activation functions are RELU: max (0, x), Tanh, Leaky RELU, etc.

3.4.Pooling Layer:

The pooling layer is key to making sure that the subsequent layers of the CNN are able to pick up larger-scale detail than just edges and curves. It does this by merging pixel regions in the convolved image together (shrinking the image) before attempting to learn kernels on it. Effectively, this stage takes another kernel, say [2 x 2] and passes it over the entire image, just like in convolution. It is common to have the stride and kernel size equal i.e., a [2 x 2] kernel has a stride of 2. This example will half the size of the convolved image. The number of feature-maps produced by the learned kernels will remain the same as pooling is done on each one in turn. Thus, the pooling layer returns an array with the same depth as the convolution layer. The figure below shows the principal.

Max pooling is the most common type of pooling operation, which extracts patches from the input feature maps, outputs the maximum value of every patch, and discards all other values (Fig.). In practice, a max pooling with a filter of size 2 2 and a stride of 2 is commonly used. The in-plane dimension of feature maps is down sampled by a factor of

2. The depth dimension of feature maps, unlike the height and width dimensions, remains constant.

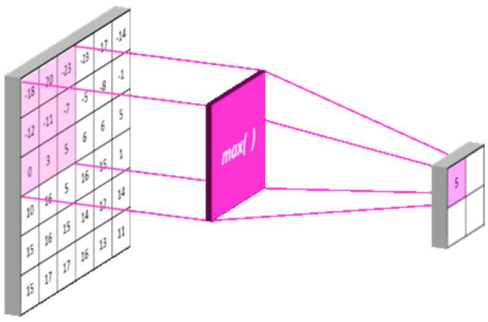

Figure 13. Max-pooling

3.5.Global Average Pooling:

A global average pooling operation is also worth mentioning. A global average pooling is an extreme type of down sampling in which a feature map with dimensions of height width is down sampled into a 1 1 array by simply taking the average of all the elements in each feature map while retaining the depth of feature maps. Typically, this operation is performed only once before the fully connected layers. The following are the benefits of using global average pooling: (1) reduces the number of learnable parameters and (2) allows the CNN to accept variable-size inputs.

3.6.Flattening:

The final convolution or pooling layer's output feature maps are typically flattened, that is, transformed into a one-dimensional (1D) array of numbers (or vector), and connected to one or more fully connected layers, also known as dense layers, in which every input is connected to every output by a learnable weight. Once the features extracted by the convolution layers and downsampled by the pooling layers are formed,

they are transferred to the network's final outputs, such as the probabilities for each class in classification tasks, by a subset of fully connected layers. Typically, the final fully connected layer has the same number of output nodes as the number of classes. Following each fully connected layer is a nonlinear function, such as.

3.7.Fully Connected Layers:

It takes the input from the previous layer and computes the final classification or regression task.

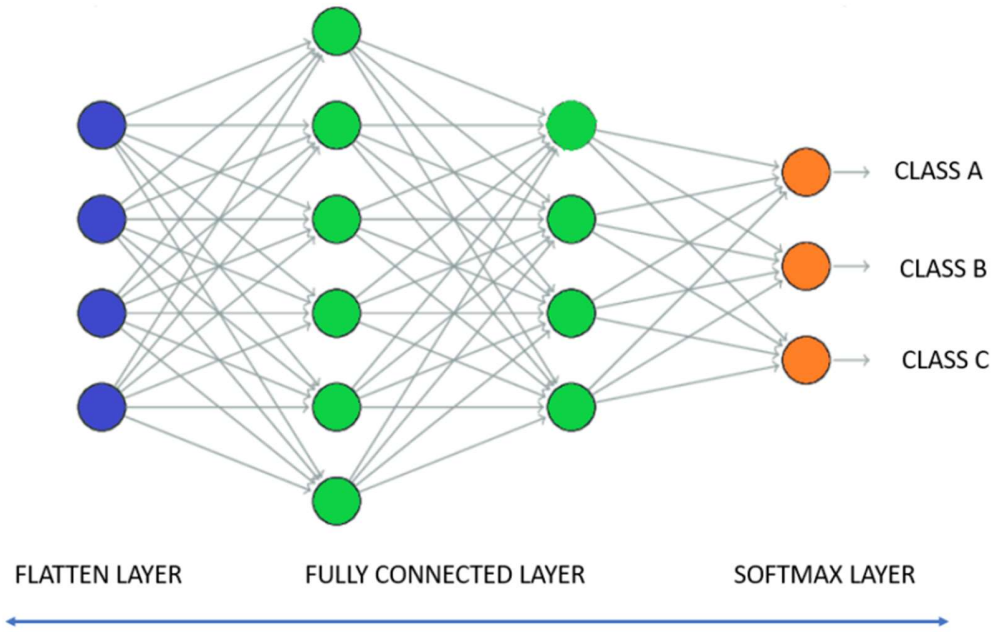

Figure 14. FC layer

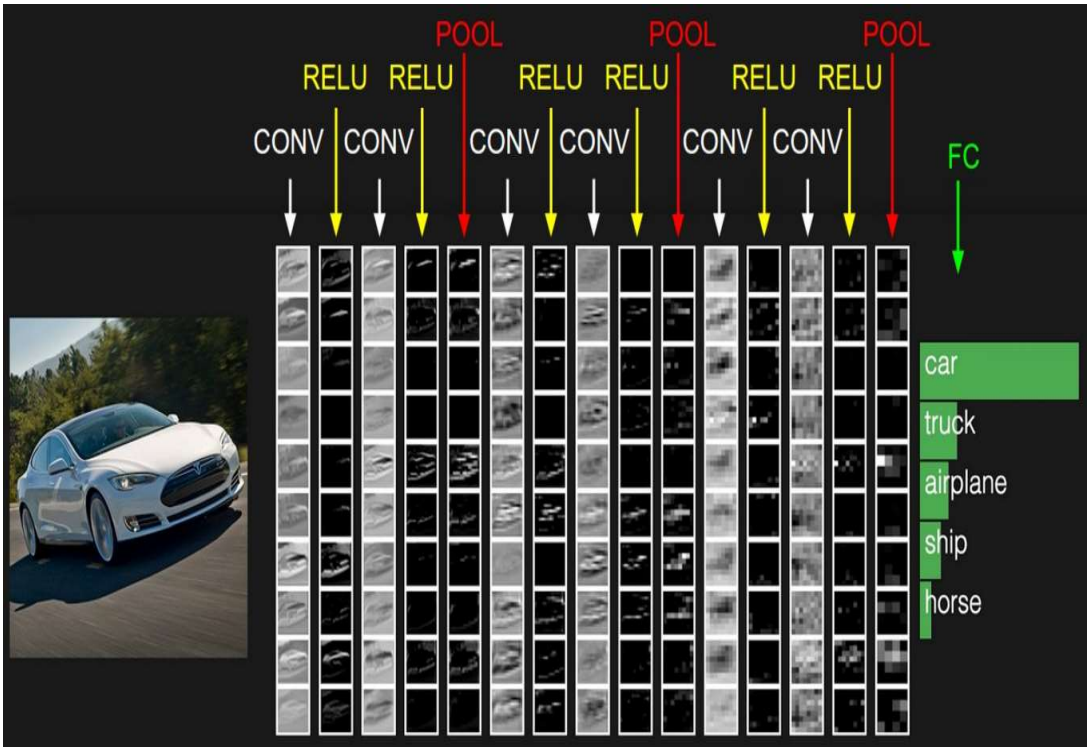

Figure 15. Layers working method

3.8.Dropout Layers:

The previously mentioned fully-connected layer is connected to all weights in the previous layer - this can be a very large number. As such, an FC layer is prone to *overfitting* meaning that the network won't generalize well to new data. There are a number of techniques that can be used to reduce overfitting though the most commonly seen in CNNs is the dropout layer, proposed by Hinton. As the name suggests, this causes the network to 'drop' some nodes on each iteration with a particular probability. The *keep probability* is between 0 and 1, most commonly around 0.2-0.5 it seems. This is the probability that a particular node is dropped during training. When back propagation occurs, the weights connected to these nodes are not updated. They are readded for the next iteration before another set is chosen for dropout.

3.9.Output Layer:

The output from the fully connected layers is then fed into a logistic function for classification tasks like sigmoid or SoftMax which converts the output of each class into the probability score of each class [12].

4. Training a network:

The process of training a network entail locating kernels in convolution layers and weights in fully connected layers that minimize discrepancies between output predictions and supplied ground truth labels on a training dataset. The backpropagation algorithm is a popular approach for training neural networks in which the loss function and the gradient descent optimization algorithm play important roles. A loss function on a training dataset calculates a model's performance under specific kernels and weights, and learnable parameters, namely kernels and weights, are updated according to the loss value using an optimization algorithm such as backpropagation and gradient descent, among others.

4.1.Loss function:

A loss function, also known as a cost function, measures the compatibility between the network's forward propagation output predictions and provided ground truth labels. Cross entropy is a commonly used loss function for multiclass classification, although mean squared error is often used for regression to continuous values. One of the hyperparameters is the type of loss function, which must be determined based on the jobs.

4.2.Gradient descent:

Gradient descent is a popular optimization approach that iteratively adjusts the network's learnable parameters, such as kernels and weights, to minimize loss. The gradient of the loss function indicates the direction in which the function has the steepest rate of rise, and each learnable parameter is updated with an arbitrary step size set by a hyperparameter termed learning rate. The gradient is a partial derivative of the loss with respect to each learnable parameter, and a single parameter update is written as follows:

$$w := w - \alpha * \frac{\partial L}{\partial w} \qquad \text{(II.40)}$$

Where w represents each parameter that can be learned, α indicates the learning rate, and L is the loss function. It is noteworthy that in practice, one of the most crucial hyperparameters to set before the training begins is a learning rate. In reality, the gradients of the loss function with respect to the parameters are calculated using a smaller subset of the training dataset known as mini-batch, and then applied to the parameter updates for reasons like memory constraints. Mini-batch gradient descent (SGD), sometimes known as stochastic gradient descent (SGD), is the name of this technique, and the mini-batch size is also a hyperparameter. Additionally, other enhancements to the gradient descent technique, including SGD with momentum, RMSprop [13].

5. Training on a small dataset:

Transfer learning, which involves pretraining a network on a very large dataset, like ImageNet, then reusing it and applying it to the specified task of interest, is a popular and efficient method for training a network on a small dataset. A fixed feature extraction method is a way to keep the remaining network, which is made up of a sequence of convolution and pooling layers and is known as the convolutional base, as a fixed feature extractor while removing FC layers from a pretrained network. Any machine learning classifier, including the common FC layers and random forests and support vector machines, can be put on top of the fixed feature extractor in this case, resulting in training limited to the added classifier on a given dataset of interest. A fine-tuning method, which

is more often applied to radiology research, is to not only replace FC layers of the pretrained model with a new set of FC layers to retrain them on a given dataset, but to fine-tune all or part of the kernels in the pretrained convolutional base by means of backpropagation. FC, fully connected.

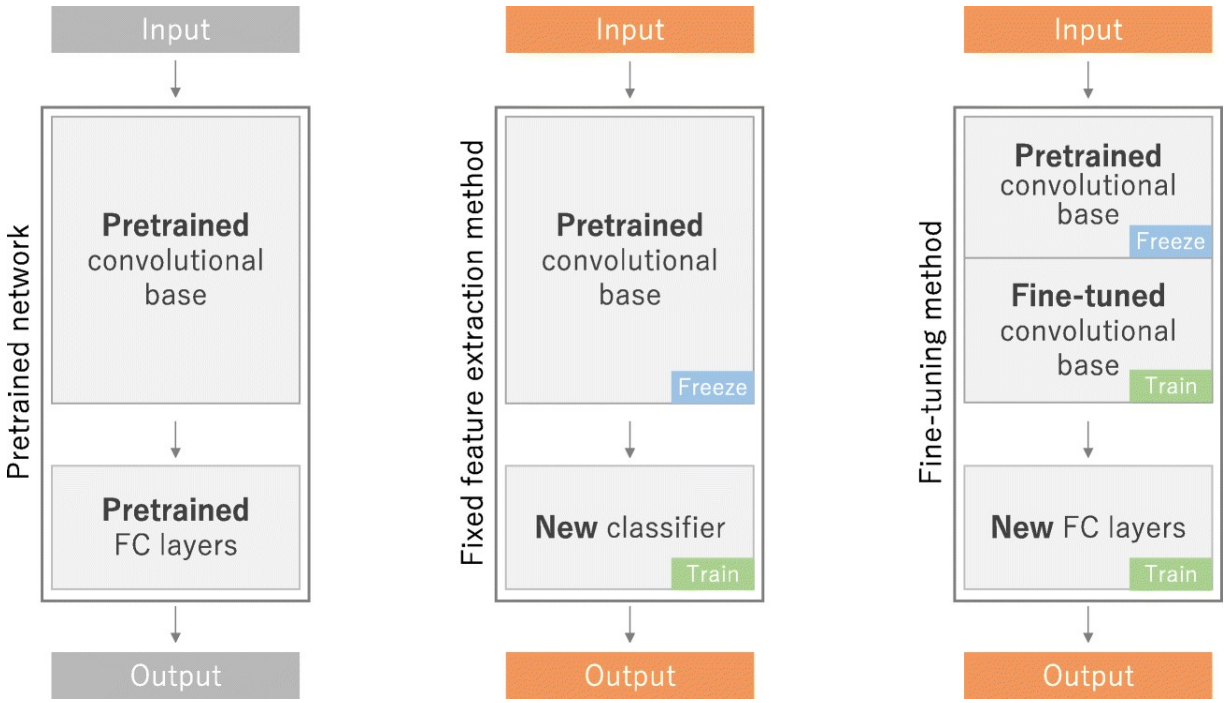

Figure 16. Training on a small dataset

5.1. Advantages of Convolutional Neural Networks (CNNs):

- Good at detecting patterns and features in images, videos, and audio signals.
- Robust to translation, rotation, and scaling invariance.
- End-to-end training, no need for manual feature extraction.
- Can handle large amounts of data and achieve high accuracy.

5.2. Disadvantages of Convolutional Neural Networks (CNNs):

- Computationally expensive to train and require a lot of memory.
- Can be prone to overfitting if not enough data or proper regularization is used.
- Requires large amounts of labeled data.
- Interpretability is limited, it's hard to understand what the network has learned [14].

Conclusion:

For contemporary visual identification tasks, Convolutional Neural Net is a prominent deep learning approach. CNN, like all deep learning algorithms, is highly dependent on the amount and quality of training data. CNNs can outperform humans at visual recognition tasks when given a well-prepared dataset. However, they are still susceptible to visual distortions like as glare and noise, with which humans can manage. CNN theory is still evolving, and researchers are aiming to provide it with qualities such as active attention and online memory, which will allow CNNs to evaluate novel items that are significantly different from what they were trained on. This more closely resembles the mammalian visual system, paving the way for a smarter artificial visual recognition system.

Chapter III:
Results and interpretations

Chapter III: Results and interpretations

Introduction:

By using the techniques outlined, we have focused on different outcomes. First of all, our objectives are to create a database in form of wav file. For that, we have used Matlab instructions (Convolutional neural network) to determine the recognition rate of the first ten Arabic digits in order to evaluate the databases' ability to display words and phrases.

We apply the following actions to accomplish the development of our recognition system.

- **Create a learning group:** each vocabulary element is recorded several times and has been named the corresponding word on the sound.
- **Auto analysis:** recorded signals are converted into series of vector characteristics (MFCC, LPC or PLP coefficients).
- **Definition of CNN models**: to form a deep learning model that detects the presence of voice commands in a voice.
- **Training models**: train a network of convolutive neurons to recognize a given set of commands.
- **Define the recognition task:** to define the grammatical rules which should be added or followed to recognize and to evaluation of performance in the test group.

1. Organisation of the database:

1.1.Organization of the corpus:

The corpus contains 20 syntactically and semantically valid Arabic sentences. Which have been verified by linguists from The University of Laghouat.

1) تَصْحِيحٌ
2) تَأْكِيدٌ
3) اَلرَّجَاءُ اِخْتِيَارَ لُغَتِكُمْ اَلْمُفَضَّلَةِ
4) اَلْفَرَنْسِيَّةَ
5) اَلْعَرَبِيَّةَ
6) اَلرَّجَاءُ إِدْخَالَ رَمْزِكُمْ اَلسِّرِّيِّ
7) اَلرَّجَاء اِخْتِيَارَ عَمَلِيَّةٍ
8) اَلسَّحْبُ
9) طَلَبُ اَلرَّصِيدِ
10)اَلْخُرُوج
11)اَلْمُتَابَعَةِ
12)اَلرَّجَاءُ اِخْتِيَارَ اَلْمَبْلَغِ
13)اَلرَّجَاءِ إِدْخَالَ اَلْمَبْلَغِ
14)1000 أَلْفِ
15)2000 أَلْفَانِ
16)3000 ثَلَاثَةِ آلَافِ
17)5000 خَمْسَةِ آلَافِ
18)10000 عَشَرَةِ آلَافِ
19)20000 عِشْرُونَ أَلْفًا
20)50000 خَمْسُونَ أَلْفًا

1.2.Identification of choice of speakers:

According to the TIMIT protocol (Texas Instruments Massachusetts Institute of Technology) the better way to manage speakers and organize the database, each speaker must have a unique code according to several criteria like gender(man/woman) age and

education level. Speakers must have a good standard Arabic pronunciation and vocal quality.

The speakers whom are recorded are students from Laghouat University and some of theme are outsiders.

Speakers:

- 60 different people:
  -35 female.
  -25 male.
- All the people were between the ages of 18 to 27
- The same phrase was recorded ten times by each person.
- Twenty Arabic words and phrases were all recorded.

1.3.Corpus registration phase:

1.3.1. Registration requirements:

We need to set all parameters correctly:

- The place of registration must be a normal place.
- Each recording must begin and end with silence
- Repeat recording if interrupted.
- The distance between the speaker and the sensor must be adjusted according to the speaker's voice.
- Check the recording every time to avoid saturation or low signal with a signal editor.

1.3.2. Record manipulation:

Read rate should be normal (neither fast nor slow). It is also necessary to correct the speaker if the emission level tends to weaken (tendency to speak less and less loudly).

- Input parameters: sampling rate (48 KHz), number of channels (1), 16 bit coded;
- The recording location is a normal place with noise:

• The corpus consists of the first ten digits of 20 sentences pronounced by 60 speakers who repeated these sentences 10 times (in our case we had total number of 12000 for the sentences) to have a good recognition rate.

Corpus		**speakers**	**Number of records per speaker**	**Total**
phrases	20	60	10	12000

Table 1.Table of corpus.

1.3.3. Acquisition of sound files:

We used adobe audition 2022 software to process our sound files. is a very effective tool that allowed us to read a sound file (view it, listen to it, cut out it…) or even create a new one, to make an acoustic analysis (durations, Fo, intensity).

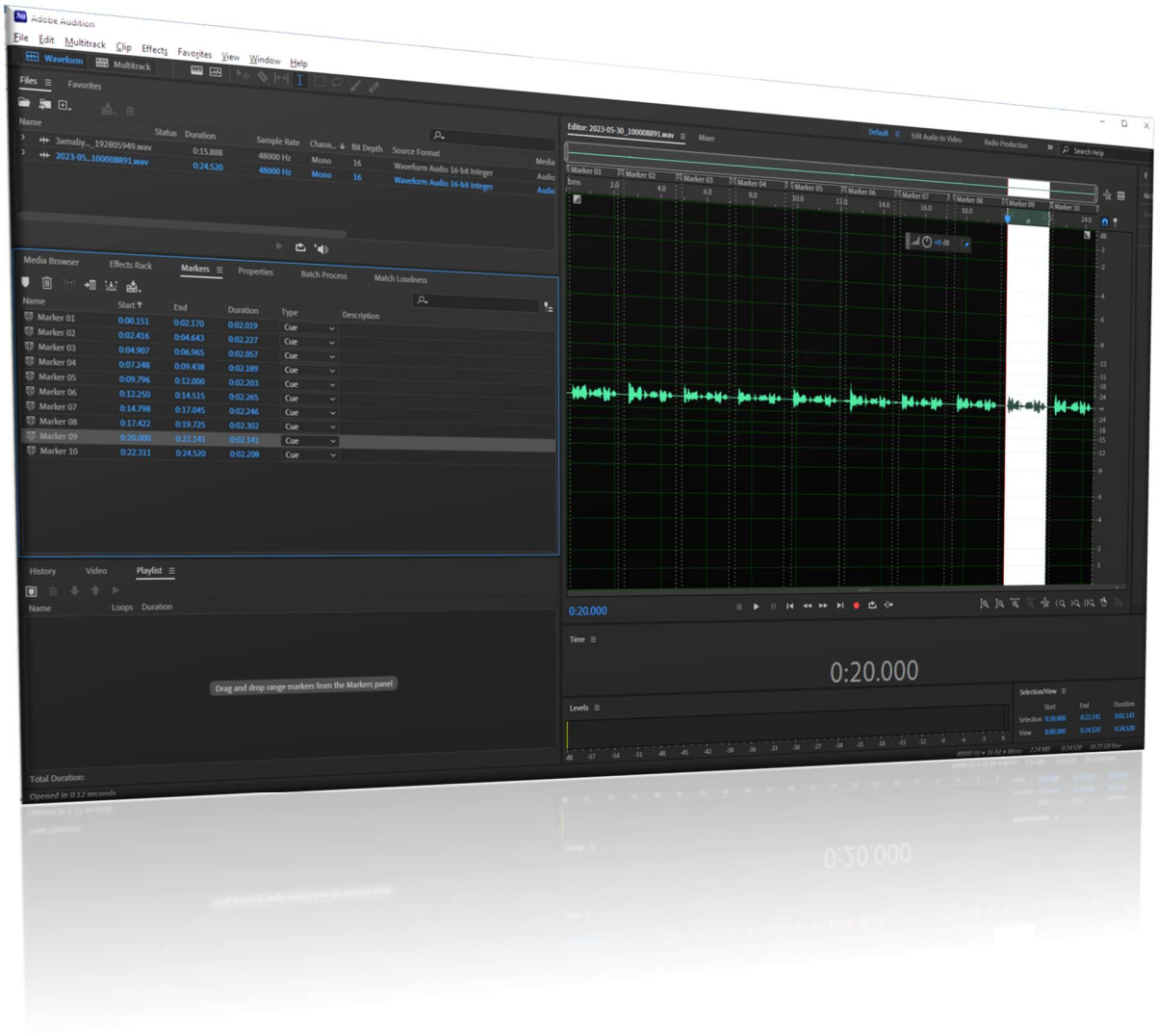

Figure 17 Adobe Audition

Each element of the vocabulary is recorded, and tagged with the corresponding word and the loc number. The result of this phase are the sound files (.wav) representing the vocabulary

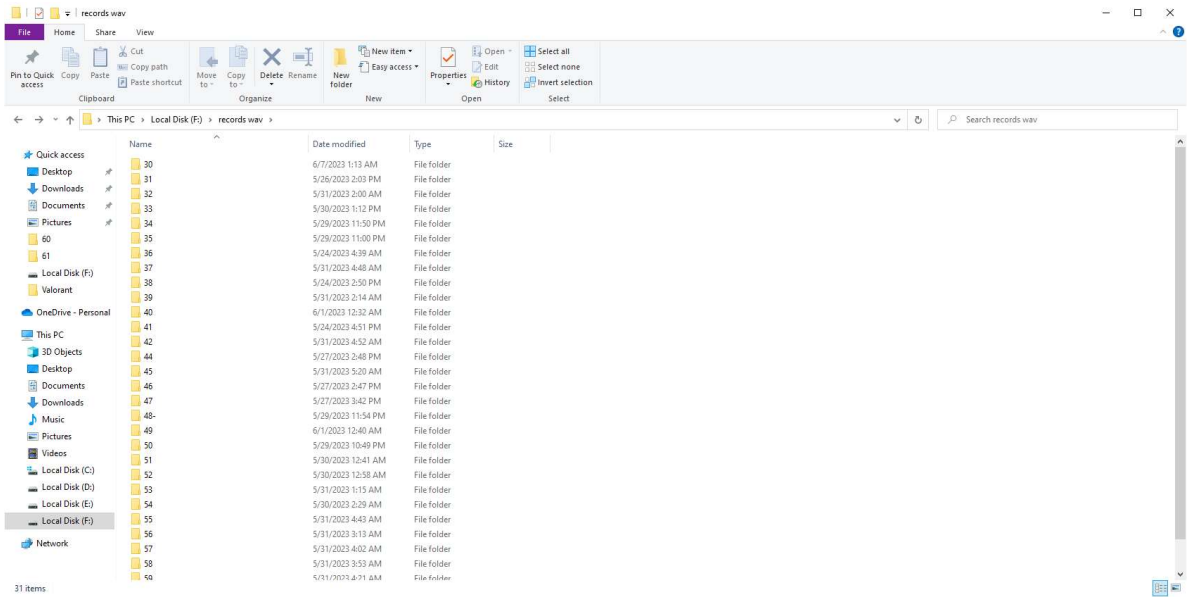

Figure 18 Loc folders

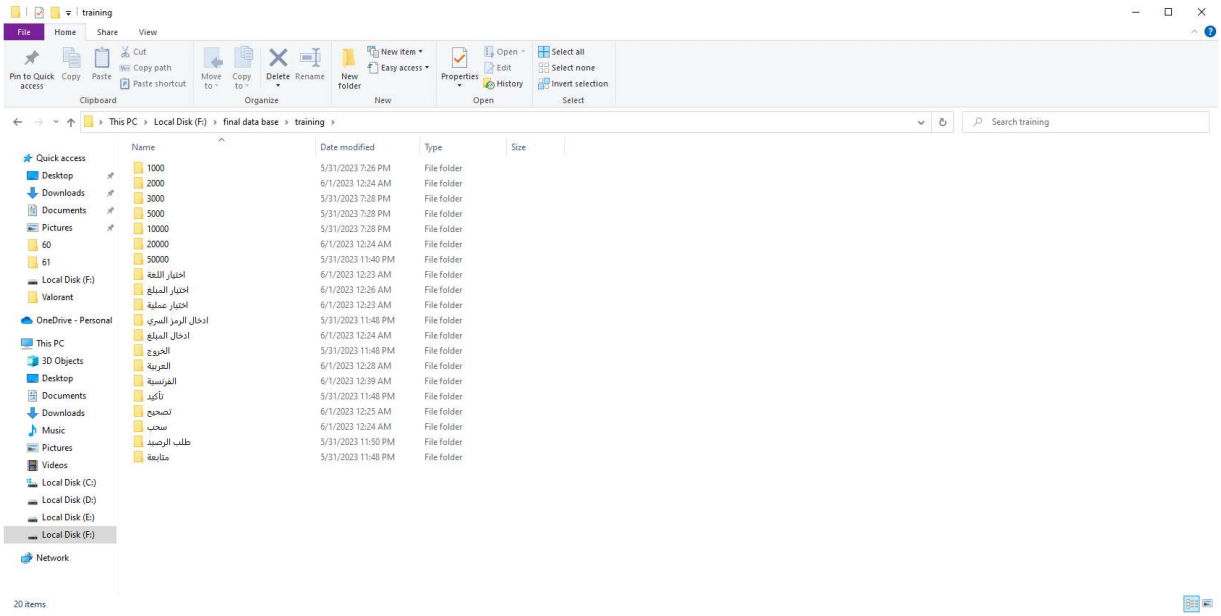

Figure 19.corpus folders

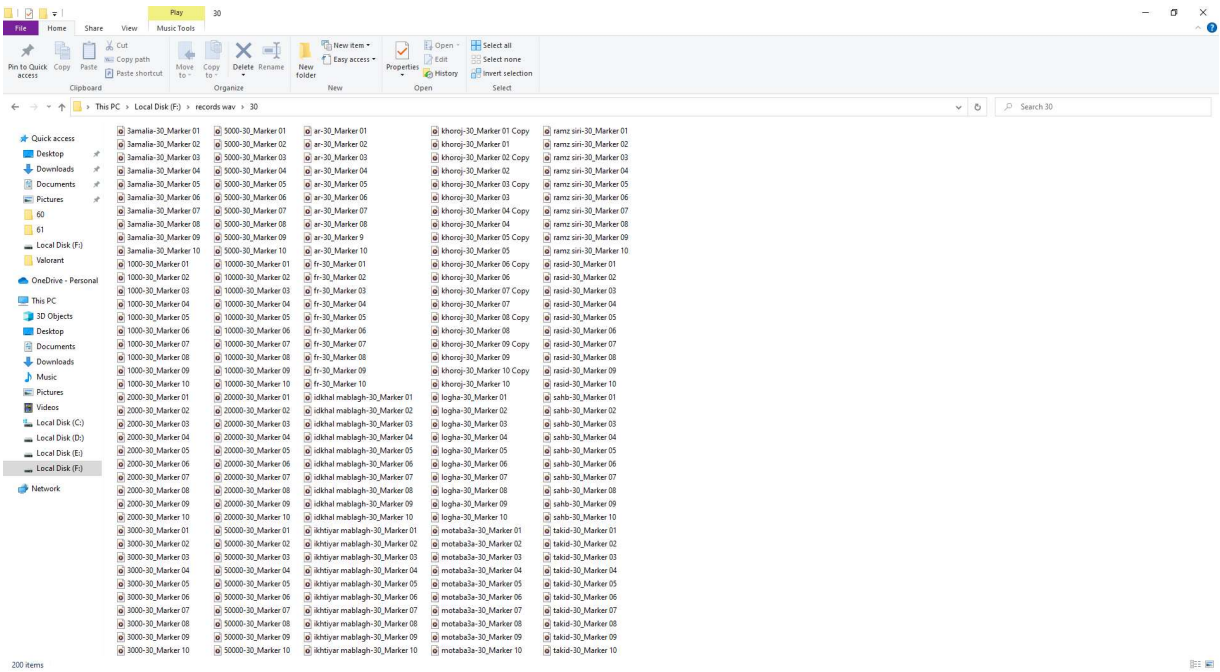

Figure 20. loc_30 Sound records

1.3.4. Final database:

Now let's see how we did separate voice records to (training /validation/test 1/test2)

- We used 60% of our dataset for training
- 30% For validation
- 10% for test 1
- And brand new 2 persons for the second test.

folder	phrases	speakers	Number of records per speaker	Total
TRAIN	20	60	6	7200
VALIDATION	20	60	3	3600
TEST 1	20	60	1	1200
TEST 2	20	2	10	400

Table 2. final DATABASE

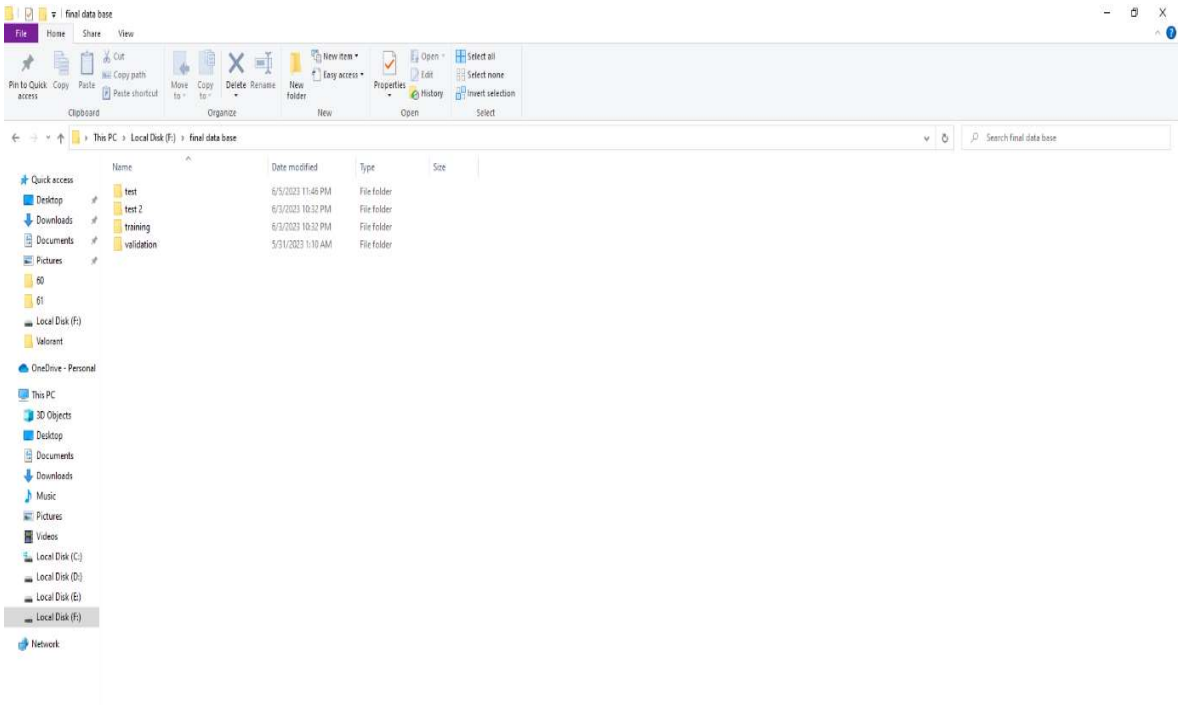

Figure 21.Final DATAbase folder

1.4.Properties of the PC station where we ran our application:

- **CPU:** Ryzen 9 5900x (12C/24T at base clock of 3.70Ghz and 4.8GHz boost)
- **GPU:** RTX 3050 (8GB DDR6 Vram and 2560 CUDA CORES)
- **RAM:** 64GB DDR5 at 4800MHz
- **ROM:**1TB team group NVMe SSD

2. Creation of neural network:

We created a simple network structure in the form of an array of layers. We used convolutive and batch flattening layers, and maps of "spatially" reduced characteristics (i.e., time and frequency) using max pooling layers. We have added a final maximum aggregation layer that aggregates the input entity map overall over time.

This reinforces (approximately) the time translation invariance in the spectral input plots, allowing the network to perform the same classification independently of the exact position of speech over time. Global aggregation also significantly reduces the number of parameters in the fully connected final layer. To reduce the network's ability to store specific data characteristics

We have added a small amount of input leakage to the last fully connected layer.

```
%CNN
numF = 13;
layers = [
imageInputLayer([numHopsnumBands])

convolution2dLayer(3,numF,'Padding','same')
BatchNormalizationLayer
```

CNN Step Matlab File.

The network is small, with only five convolutive layers and some filters. numF Controls the number of filters in convolutive layers. To increase network resolution, we tried to increase network depth by adding identical blocks of convolutive layers, batch normalization and ReLU. We can also try to increase the number of convolutive filters by increasing numF.

```
reluLayer

maxPooling2dLayer(3,'Stride',2,'Padding','same')

convolution2dLayer(3,2*numF,'Padding','same')
batchNormalizationLayer
reluLayer

maxPooling2dLayer(3,'Stride',2,'Padding','same')

convolution2dLayer(3,4*numF,'Padding','same')
batchNormalizationLayer
reluLayer

maxPooling2dLayer(3,'Stride',2,'Padding','same')

convolution2dLayer(3,4*numF,'Padding','same')
batchNormalizationLayer
reluLayer
```

```
convolution2dLayer(3,4*numF,'Padding','same')
batchNormalizationLayer
reluLayer

maxPooling2dLayer([timePoolSize,1])

dropoutLayer(dropoutProb)
fullyConnectedLayer(numClasses)
softmaxLayer];
```

Matlab file presenting the CNN layers

2.1.Training Network:

We selected training options. We used the Adam Optimizer with a small batch size of 20. He trained for 25 periods and reduced the learning rate by a factor of 20 then 100 epochs.

```
miniBatchSize = 20;%128;
validationFrequency = floor(numel(YTrain)/miniBatchSize);
options = trainingOptions('adam',...
'InitialLearnRate',3e-4, ...
'MaxEpochs',25, ...
'MiniBatchSize',miniBatchSize, ...
'Shuffle','every-epoch', ...
'Plots','training-progress', ...
'Verbose',false, ...
'ValidationData',{XValidation,YValidation}, ...
'ValidationFrequency',validationFrequency, ...
'LearnRateSchedule','piecewise', ...
'LearnRateDropFactor',0.1, ...
'LearnRateDropPeriod',20);
```

Train the network:

```
trainedNet = trainNetwork(XTrain,YTrain,layers,options);
```

2.1.1. Visualize Data:

To prepare the data for an effective formation of a network of convolutive neurons, we transformed the waveforms of speech into auditory spectrograms.

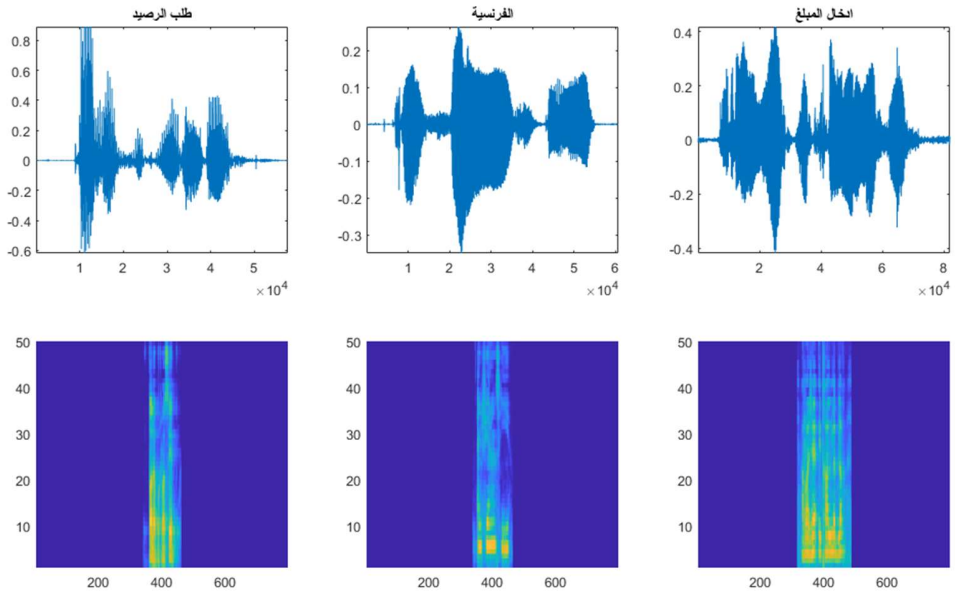

Figure 22. Auditory spectrograms of some learning samples

2.1.2. Distribution of the different classes:

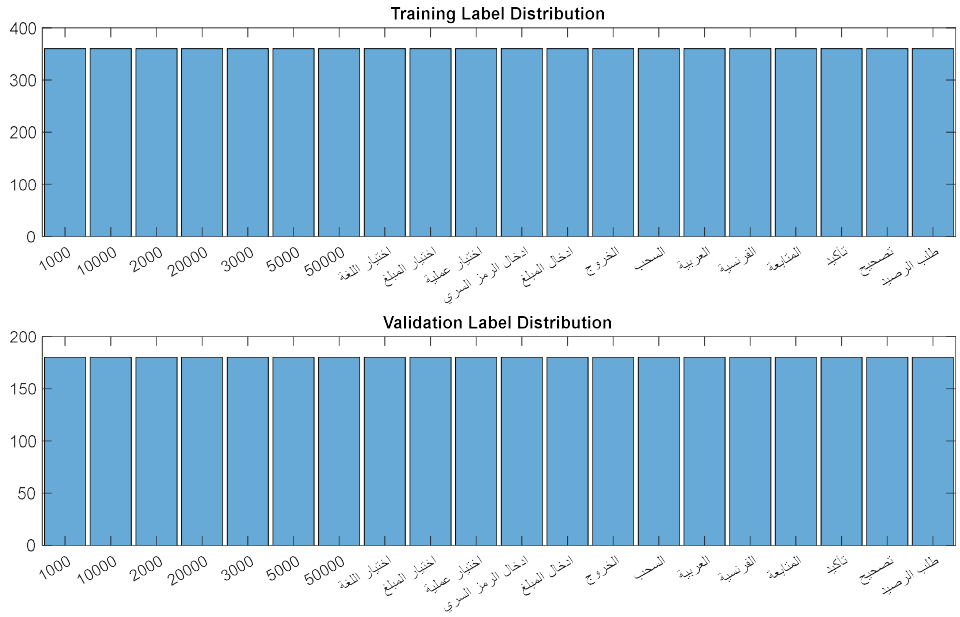

Figure 23. Distribution of the different class labels in the learning and validation packages.

From this graph we conclude that all our records are uploaded and been read successfully.

3. Results:

We conducted several tests and are delighted to share all of the results we have obtained.

3.1.Validation results:

We performed two validation tests that exhibited variations due to the difference in the number of epoch.

3.1.1. Epoch:

One epoch indicates that each sample in the training dataset has been utilized once to update the internal model parameters. An epoch can be composed of one or more batches. For instance, when an epoch consists of only one batch, it is known as the batch gradient descent learning algorithm

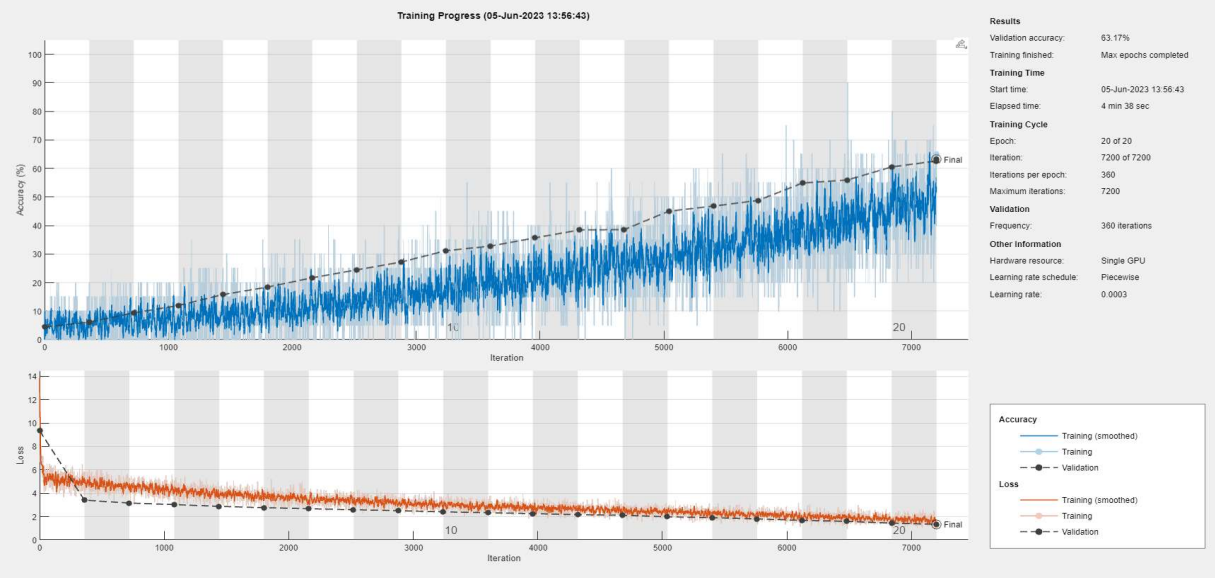

Figure 24. Results with 20 epoch

3.1.1.1. Validation results with 20 epochs:

We have noticed that when using 20 epochs, we only achieve 7200 iterations, which is considered a small number for a relatively large database therefore, the inadequate number of iterations resulting from the 20 epochs is the reason behind the less-than-desirable outcome we achieved (63% accuracy and 3.3 error)

Based on our observation, we concluded that increasing the number of epochs, and thus the number of iterations, would lead to improved results. Consequently, we conducted experiments using 30, 40, 50, 60, 70, 80, 90, and 100 epochs.

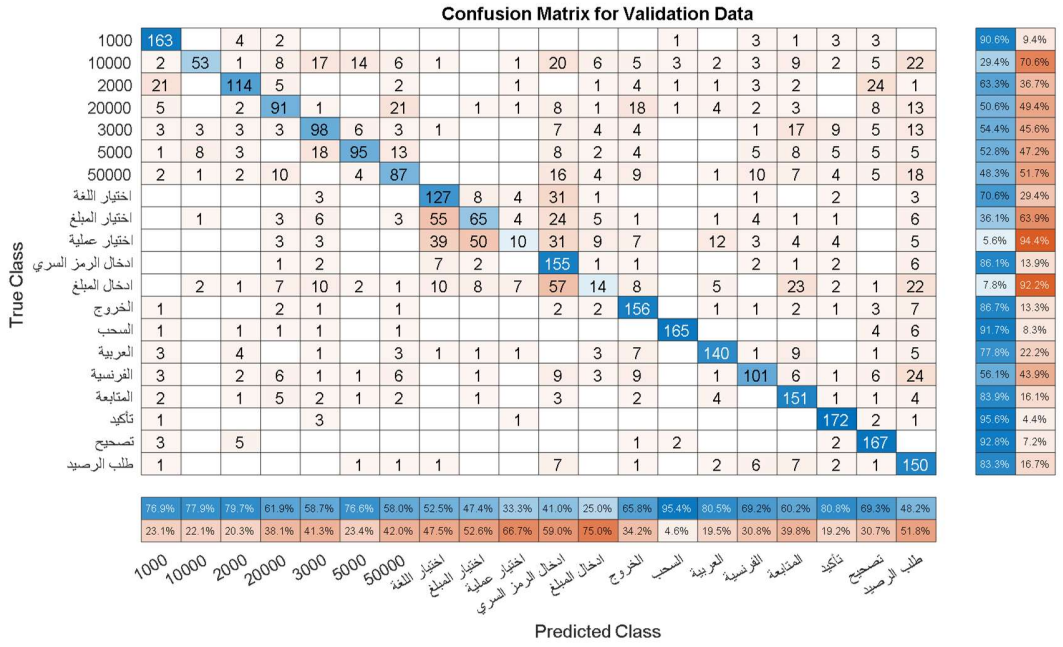

Figure 25. Confusion matrix with 20 epoch

3.1.1.2. Results with 100 epochs:

Among the different epoch configurations tested, we obtained the optimal results with 100 epochs.

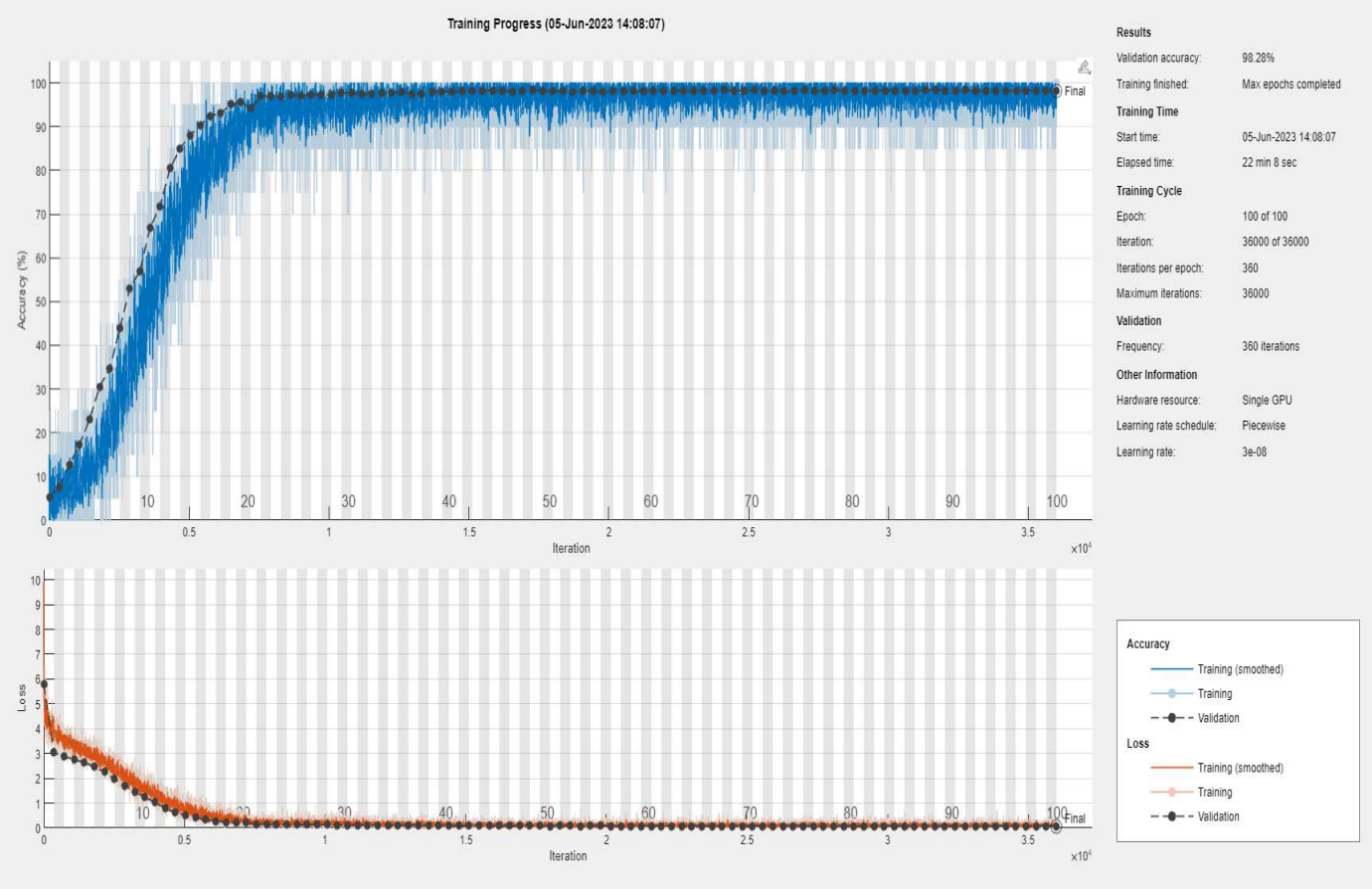

Figure 26. Results with 100 epochs

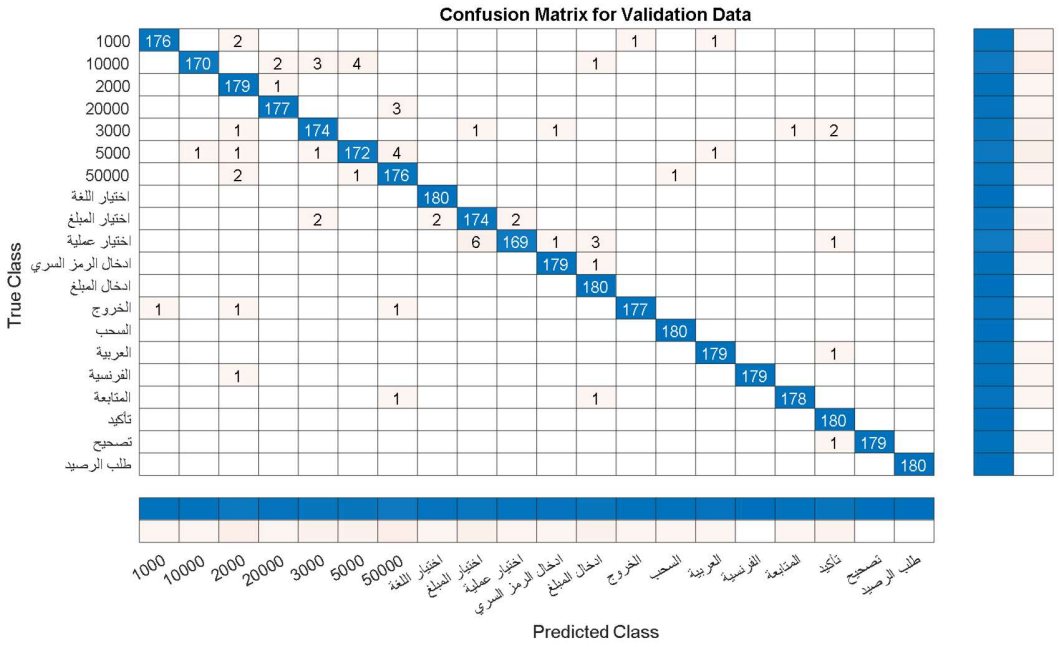

Figure 27. confusion matrix with 100 epochs

When using 100 epochs, we noticed a considerable jump in the number of iterations to 36,000, which is highly suitable for our database. This substantial increase in iterations played a pivotal role in achieving nearly perfect results:

- **Accuracy:** The results obtained on the test data shown in Figure (6) above gave us a rate of 98.28%. And the results on the training data gave us a rate of 98.28%.
- **Loss**: The results obtained on the resulting rate of loss represented in Figure (6)above gave us a rate of 0.04%.

3.2.Test results:

- Test results of the same speakers:

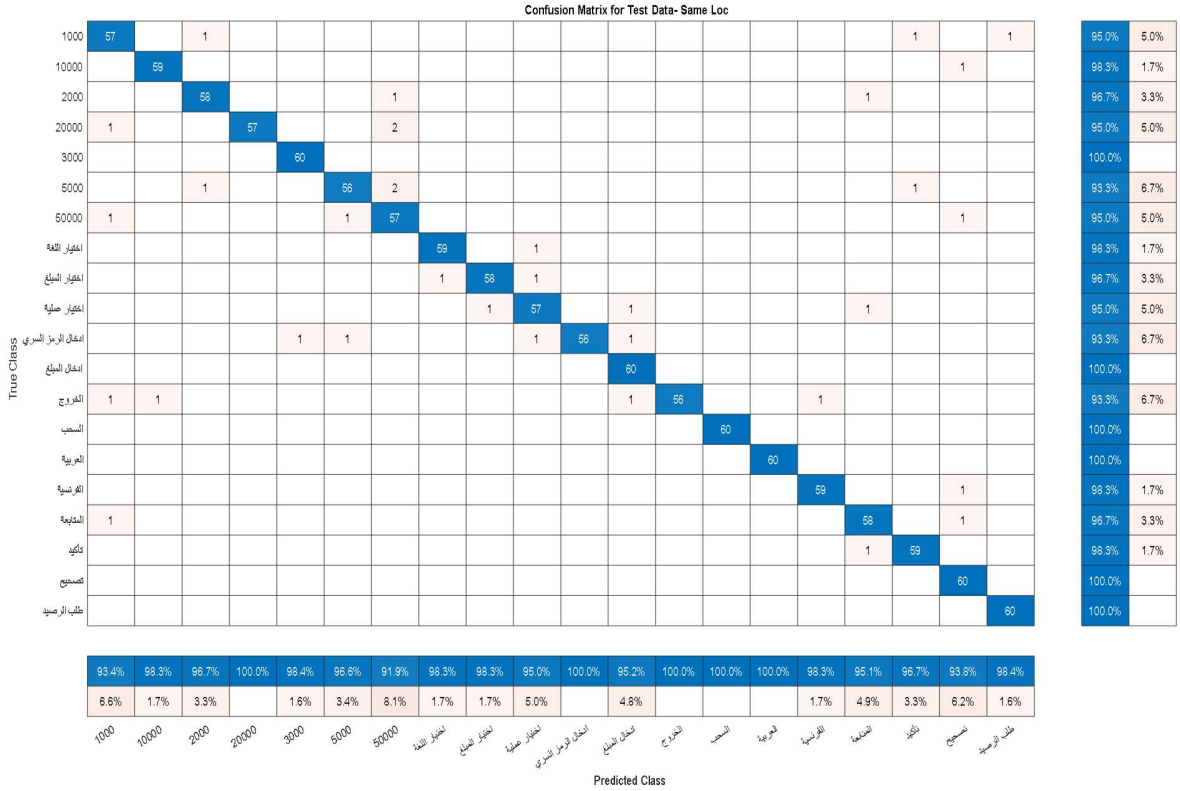

Figure 28. Results of the same LOC

In this test we used one phrase (one record) from every speaker we record earlier in every word they recorded. Which left us with 1200 record.

In this table like we see the result with accuracy of 97% and 0.08% loss. Witch indicates to a good result.

- Test results of new speakers:

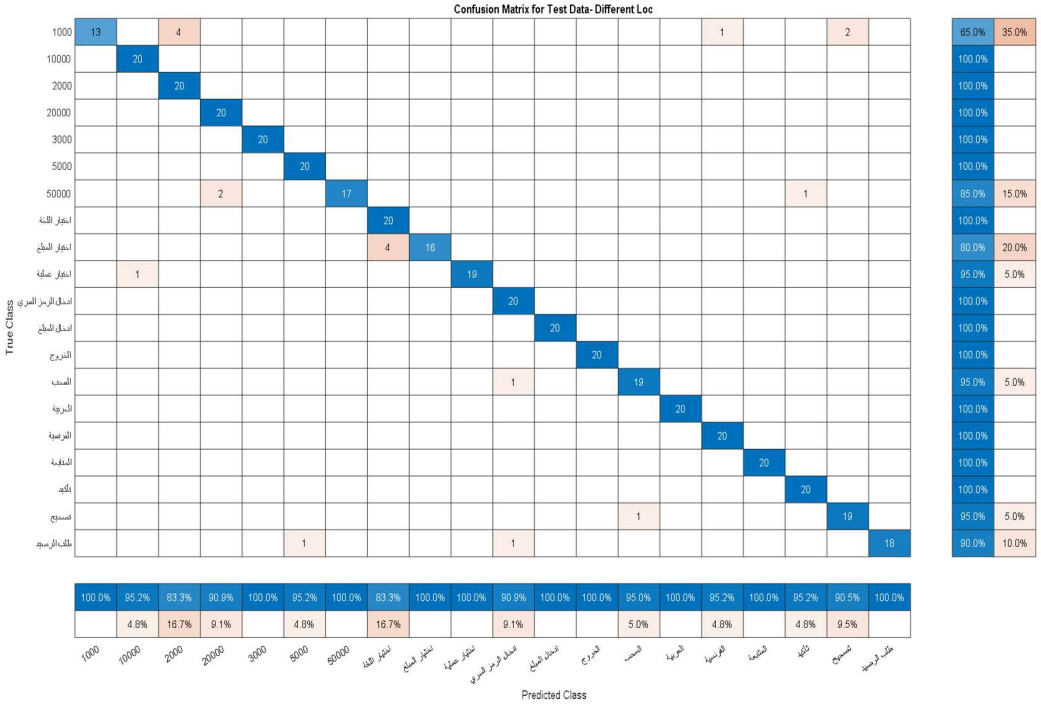

Figure 29. Results Of new LOC

In this second test we confirmed the efficiency of our code using deferent speakers that never been recorded before. With that said we've reached a level of accuracy of 97% and 0.07% loss.

General Conclusion

General Conclusion:

The spoken word is the primary method of communication in any human society, and of course, it is also the most natural method one. it is also true that it is semantically easier. Whom developing speech recognition systems, to facilitate human-machine communication and to enable machines to operate in natural language is a challenge.

The work we are undertaking here aligns with the goal of establishing a human-machine dialogue system. It involves speech recognition in the field of communications.

Within the scope of this thesis, we have focused specifically on automatic speech recognition (ASR) applied to recorded voices. We have utilized MFCC and LPC analysis techniques for automatic recognition. Our choice has also involved selecting the most appropriate tools such as CNN, MATLAB, etc. for ASR.

Our contributions to this topic can be summarized as follows:

- Acquisition and development of a database consisting of a corpus of syntactically and semantically correct words in classical Arabic. They were evaluated by linguists from the University of Laghouat.
- Statistical modeling using convolutional neural networks (CNN) for speech recognition.
- Development of a reference system for ASR based on CNN modeling using MATLAB.

The results obtained leads us to conclude that the CNN method is highly effective with voice treatment and ASR.

References:

[1] S. Rapuano and F. Harris, “An introduction to FFT and time domain windows,” IEEE Instrumentation and Measurement Society, vol. 10, no. 6, Dec. 2007, pp. 32-44.

[2] W. A. Mahmoud, “Quantization Techniques for The Classification and Recognition of Speech Signals”, Ph.D. Thesis, University of Swansea, England, 1986.

[3] Oday Kamil Hamid, “Speech Sound Coding Using Linear Predictive Coding (LPC)”, Journal of Information, Communication, and Intelligence Systems (JICIS), Volume 3, Issue 1, May 2017.

[4] F. Itakura, “Minimum Prediction Residual Principle Applied To Speech Recognition”, IEEE Trans. Acoust, Speech and Signal Process, Vol. ASSP-23, Feb. 1975, pp. 67-72.

[5] Rabiner, Lawrence R., and B. H Juang. Fundamentals of Speech Recognition. Englewood Cliffs, N.J.: PTR Prentice Hall, 1993.

[6] Maheshkumar B. Landge and R.R. Deshmukh and P.P. Shrishrimal “Analysis of Variations in Speech in Different Age Groups using Prosody Technique», International Journal of Computer Applications (0975 – 8887), Volume 126 – No.1, September 2015.

[7] H. Hermansky, “Perceptual linear predictive (PLP) analysis of speech,” Acoustical Society of America Journal, vol. 87, pp.1738 – 1752, Apr. 1990.

[8] Lei Xie, Zhi-Qiang Liu, “A Comparative Study of Audio Features for Audio to Visual Cobversion in MPEG-4 Compliant Facial Animation,” Proc. of ICMLC, Dalian, 13-16 Aug-2006.

[9] Convolutional Neural Networks – Basics by: Rob Robinson

[10] Introduction to Convolution Neural Network by: Geeks for Geeks; 11/04/2023.

[11] Convolutional neural networks: an overview and application in radiology by : Rikiya Yamashita, Mizuho Nishio and Kaori Togashi.15/04/2023.

[12] LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444.

[13] Russakovsky O, Deng J, Su H et al (2015) ImageNet Large Scale Visual Recognition Challenge. Int J Comput Vis 115:211–252.

[14] Fukushima K (1980) Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybern 36:193–202.

Abstract:

Automatic speech recognition has been an active field of study since the 1950s, which explains its richness but also its difficulty. It involves the collaboration of multiple disciplines and techniques. The complexity of the speech signal, resulting from the interaction between sound production and perception by the ear, contributes to the challenge of automatic speech recognition, which has become a highly interesting research topic.

The objective of this thesis is the acquisition and implementation of a database consisting of 20 phrases divided in to two categories connected words and separated words, along with a corpus of syntactically and semantically correct sentences. This database was recorded under real conditions, and the acoustic analysis of this database was performed using the MFCC method, providing us with a series of input vectors for the implemented Automatic Speech Recognition (ASR) system. This system is based on Convolutional Neural Networks. Evaluating the performance of the ASR system using the database analysis method will highlight the influence of parameterization.

ملخص :

كان التعرف التلقائي على الكلام مجالًا نشطًا للدراسة منذ الخمسينيات من القرن الماضي، مما يفسر ثرائه ولكن أيضًا صعوبته. ويشمل التعاون بين عدة تخصصات وتقنيات. يساهم تعقيد إشارة الكلام، الناتجة عن التفاعل بين إنتاج الصوت وإدراك الأذن، في تحدي التعرف التلقائي على الكلام، والذي أصبح موضوعًا بحثيًا مثيرًا للاهتمام للغاية.

الهدف من هذه الأطروحة هو الحصول على وتنفيذ قاعدة بيانات تتكون من 20 عبارة مقسمة إلى مجموعتين كلمات متصلة وكلمات منفصلة، إلى جانب مجموعة من الجمل الصحيحة من الناحية النحوية والدلالية. تم تسجيل قاعدة البيانات هذه في ظل ظروف حقيقية، وتم إجراء التحليل الصوتي لقاعدة البيانات هذه باستخدام طريقة MFCC، مما يوفر لنا سلسلة من جهات الإدخال لنظام التعرف التلقائي على الكلام (ASR) المنفذ. يعتمد هذا النظام على الشبكات العصبية الالتفافية. سيسلط تقييم أداء نظام ASR باستخدام طريقة تحليل قاعدة البيانات على تأثير الإعدادات المطبقة.