



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΤΜΗΜΑ ΙΣΤΟΡΙΑΣ ΚΑΙ ΦΙΛΟΣΟΦΙΑΣ ΤΗΣ ΕΠΙΣΤΗΜΗΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΦΙΛΟΛΟΓΙΑΣ
ΤΜΗΜΑ ΨΥΧΟΛΟΓΙΑΣ

Παρασκευοπούλου Γεωργία

A.M.: 17M13

Αναγνώριση Συναισθήματος με χρήση Βαθιάς Μάθησης και Πρωτότυπων Τεχνικών Επαύξησης Δεδομένων

Διπλωματική εργασία
για τη λήψη μεταπτυχιακού διπλώματος ειδίκευσης από το
Διατμηματικό Πρόγραμμα Μεταπτυχιακών Σπουδών στη Γνωσιακή Επιστήμη

Συμβουλευτική επιτροπή:
Περαντώνης Σταύρος, Διευθυντής Ερευνών, ΕΚΕΦΕ Δημόκριτος
Σπύρου Ευάγγελος, Επίκουρος Καθηγητής, Παν. Θεσσαλίας
Καράλη Ιζαμπώ, Επίκουρη Καθηγήτρια, ΕΚΠΑ

Αθήνα,
Οκτώβριος 2021

Εγκρίνεται η διπλωματική εργασία

Περαντώνης Σταύρος.....

Σπύρου Ευάγγελος.....

Καράλη Ιζαμπώ.....

Contents

Abstract	7
Περίληψη	9
Speech Emotion Recognition	11
Introduction	11
1. Emotions	13
1.1 Databases	14
1.2 Speech Signal Representation	14
1.3 Short-term audio processing	15
1.3.1 Framing and Windowing	15
1.3.2 Extraction of Prosodic features	16
Zero Crossing Rate	17
Short time Energy	17
Pitch	18
1.3.3 Spectral features	19
MelFrequency Cepstral Coefficients (MFCC)	19
Linear Prediction Cepstral Coefficients (LPCC)	21
Formant Features	21
1.3.4 Other features	22
1.3.5 Feature selection and dimension reduction	22
1.4 Related Work	23
Chapter summary	26
2. Convolutional Neural Networks	27
2.1 Convolutional layers	28
2.2 Pooling Layers	29
2.3 Normalizing Layers	29
2.3.1 Input-Batch-Normalization	30
2.3.2 Local Response Normalization Layers	31
2.4 Fully-Connected Layers	31
2.5 Dropout	32
2.6 Weight Initialization	33
Chapter summary	36
3. Method and Experiments	37
3.1 Emotional speech dataset	37
3.2 Data preprocessing and Augmentation	37
Noise	38

Shifting Time	39
Change Pitch	39
Change Speed	39
3.3 Inputs to network	40
3.4 Experiments with EMOVO dataset	43
3.4.1 Network Architecture	43
3.4.2 Training Process	44
3.4.3 Augmentation Choice	46
3.4.4 Models	47
3.4.5 Results	48
3.5 Experiments with other databases and cross-language testing	52
3.6 Tools and Libraries	53
3.7 Discussion and future work	54
Chapter summary	55
References	57

Abstract

Emotion recognition is quite important for various applications related to human-computer interaction or for understanding the user's mood in specific tasks. In general, a person's emotion is recognized by analyzing facial expressions, gestures, posture, speech or physiological parameters such as those occurring from electroencephalograms, electrocardiograms, etc. However, in many cases, the visual information is not available or appropriate, while the measurement of physiological parameters is difficult and requires specialized, expensive equipment. As a result, speech is probably the best alternative.

The typical machine learning techniques used for this purpose extract a set of linguistic features from the data, which are then used to train supervised learning models. In this thesis, a Convolution Neural Network (CNN) is proposed, which, unlike traditional approaches, detects only the important features of raw data entered into it. It is worth noting that the architecture of a CNN is analogous to the connectivity of the neurons of the human brain and inspired by the organization of the visual cortex.

The inputs to the neural network are the spectrograms that are extracted from audio signals. For the optimal performance of the algorithm, data augmentation techniques of the original data are applied such as adding noise, shifting of the audio signal, and changing its pitch or its speed. Finally, methods against overfitting are applied, such as dropout and local response normalization layers, the operation of which is inspired by lateral inhibition of the neurons of the human brain. Our approach outperformed previous work, without being established as a considerably language-independent one.

Περίληψη

Η αναγνώριση του συναισθήματος είναι αρκετά σημαντική για διάφορες εφαρμογές σχετικές με την αλληλεπίδραση ανθρώπου - υπολογιστή ή την κατανόηση της διάθεσης του χρήστη σε συγκεκριμένα task. Γενικά, το συναίσθημα ενός ανθρώπου αναγνωρίζεται αναλύοντας εκφράσεις του προσώπου, χειρονομίες, τη στάση του σώματος, την ομιλία ή φυσιολογικές παραμέτρους όπως αυτές προκύπτουν από ηλεκτροεγκεφαλογράφημα, ηλεκτροκαρδιογράφημα κα. Ωστόσο, σε πολλές περιπτώσεις οι οπτικές πληροφορίες δεν διαθέσιμες ή κατάλληλες, ενώ η μέτρηση των φυσιολογικών παραμέτρων είναι δύσκολη, δύσχρηστη και απαιτεί εξειδικευμένο ακριβό εξοπλισμό. Συνεπώς, η ομιλία ίσως είναι η καλύτερη εναλλακτική.

Οι συνηθισμένες τεχνικές μηχανικής μάθησης που χρησιμοποιούνται για το σκοπό αυτό εξάγουν ένα σύνολο γλωσσολογικών χαρακτηριστικών από τα δεδομένα, τα οποία χρησιμοποιούνται στη συνέχεια για την εκπαίδευση μοντέλων επιβλεπόμενης μάθησης (supervised learning). Στη διπλωματική αυτή χρησιμοποιείται ένα μοντέλο Συνελικτικού Νευρωνικού Δικτύου (Convolution Neural Network - CNN) που σε αντίθεση με τις παραδοσιακές προσεγγίσεις ανιχνεύει μόνο τα σημαντικά χαρακτηριστικά των δεδομένων που εισάγονται σε αυτό. Αξίζει να σημειωθεί, πως η αρχιτεκτονική ενός CNN είναι ανάλογη με τη συνδεσιμότητα των νευρώνων του ανθρώπινου εγκεφάλου και εμπνευσμένη από την οργάνωση του οπτικού φλοιού.

Χρησιμοποιούνται τρία σύνολα ηχητικών δεδομένων (EMOVO, SAVEE, Emo-DB), από όπου εξάγονται τα αντίστοιχα φασματογραφήματα (spectrograms), τα οποία με τη σειρά τους χρησιμοποιούνται ως είσοδοι στο νευρωνικό δίκτυο. Για τη βέλτιστη απόδοση του αλγορίθμου εφαρμόζονται πρωτότυπες τεχνικές επαύξησης (data augmentation) των αρχικών δεδομένων πέραν της συνηθισμένης πρόσθεσης noise, όπως μετατόπιση του ηχητικού σήματος, αλλαγή της οξύτητας και της ταχύτητας του. Τέλος, χρησιμοποιούνται μέθοδοι καταπολέμησης της υπερπροσαρμογής (overfitting) όπως το dropout και τεχνικές ενίσχυσης της γενικευσιμότητας του μοντέλου όπως πρόσθεση επιπέδων κανονικοποίησης τοπικής απόκρισης (local response normalization layers), η λειτουργία των οποίων είναι εμπνευσμένη από την πλευρική αναστολή (lateral inhibition) των νευρώνων του εγκεφάλου. Τα αποτελέσματα είναι βελτιωμένα σε σχέση με άλλες παρόμοιες μελέτες. Ωστόσο, το μοντέλο δεν υποδεικνύει ανεξαρτησία από τη γλώσσα των ηχητικών σημάτων.

Speech Emotion Recognition

Introduction

The word “emotion” is composed of the prefix “e”, which means “out” and the word “motion”, which means move. Therefore, emotions are feelings that make us move. That is, we experience them, we express them, we recognize them, and use them to understand each other and make decisions, consciously or not. It’s worth reporting that the beginning of research related to emotions seems to be 1872 when Charles Darwin published *The Expression of the Emotions in Man and Animals*. He argued that all humans, and even other animals, use similar behaviors to express their emotions. Nowadays, many psychologists agree with Darwin that certain emotions are universal to all humans, regardless of culture. These emotions are known as the “big six” anger, fear, surprise, disgust, happiness, and sadness (Ekman and Oster, 1979; Cowie & Cornelius, 2003).

According to Cowie, Douglas-Cowie, Tsapatsoulis, Votsis, Kollias, Fellenz, and Taylor (2001), when humans interact two communication channels are used. The speaker transmits explicit messages and the listener transmits implicit messages about the speaker. The first explicit channel has highly been a region of research when the second implicit one has been little studied in the last two decades. As a matter of fact, emotion recognition is a difficult task due to different ways of measuring and categorizing emotions or their dependency on various factors that make even humans misinterpret them. Nevertheless, emotion recognition is a very important part of our lives and as a result, it is extended to computers in order to enrich AI applications with emotional intelligence. There are many applications in human-computer interaction like lie detection, clinical diagnosis of schizophrenia, voice production for synthetic agents, robots, or machines that act as personal assistants. In addition, automotive environments use the information for the driver’s emotional state to apply safety strategies. Emotion recognition is also applicable in gaming when the “intelligent” video game recognizes the player’s mood and makes the game more interesting and intractable if he feels positive emotions for example or easier for negative emotions. All these applications have something in common, they take as input the user’s response or reaction, detect its emotional state, and subsequently make the appropriate decision.

Emotion recognition can be realized through various modalities. Visual signals like facial expressions, gestures or body movements, and audio signals are widely used separately or with fusion techniques accomplishing multimodal recognition. Textual features as derived from speech could enhance the performance of a system that recognizes emotions. Biosignals like Blood Volume Pulse or skin conductivity and brain signals from EEG (Song, Zheng, Song, & Cui, 2018) or functional Magnetic Resonance Imaging (fMRI) (Han, Ji, Hu, Guo & Liu, 2015) have been used in some research. While it is difficult to collect such signals out of a laboratory environment. Obviously, speech is the most available modality and sometimes is the only one, like a call center’s application. Furthermore, speech carries much information that reflects emotional content that can result in achieving high recognition rates. For that

reason, there is a field of research that aims to design systems that recognize emotions only from audio signals, that is Speech Emotion Recognition (SER). Ideally, the best SER system should be universal and robust against language or culture (Schuller, 2018).

The definition of a Speech Emotion Recognition system is a collection of methodologies that are used in order to process and classify speech signals, detecting emotions embedded in them (Akçay & Oğuz, 2020). To achieve emotion recognition, a SER system requires a supervised learning method, particularly a classifier that will be trained to recognize emotional states. For that purpose, labeled data are essential. Given a dataset of speech signals, preprocessing is necessary and various important features can be extracted in order to be passed to the classifier as inputs. Prosodic, spectral, voice quality features and features based on the Teager energy operator are common categorizations. A wide range of classifiers is commonly used such as Support Vector Machines, Hidden Markov Models, Decision Trees, or Ensemble Methods. Deep Learning techniques have recently entered the game with impressive results through learning from raw speech data.

In this work, an approach that uses deep neural networks and does not require handcrafted features is presented. It relies on a Convolutional Neural Network (CNN), whose inputs are the spectrograms of the audio signals, and its performance is enhanced through audio data augmentation. In particular, audio training samples are increased by adding noise, shifting, changing pitch, or changing speed. EMOVO, SAVEE and Emo-DB are used to train the CNN, in an attempt to affirm that deep learning is indeed able to replace typical approaches which need feature extraction. The performance of our emotion classifier surpasses the performance of similar works. Finally, cross-language testing was actualized, but the CNN approach was not accentuated as a remarkably language-independent one.

The first section of the thesis presents a general description of Emotion Recognition Systems and related work within the broader research area of emotion recognition from speech. The second section provides a theoretical background of Convolutional Neural Networks. Finally, the proposed approach in detail, results, and discussion are presented in the third section.

1. Emotions

Successful implementation of a SER needs an appropriate definition and representation model of emotion. However, there are more than ninety definitions of emotion according to Plutchik (2001). This confirms the definition of emotion as an open problem in psychology. In practice, there are two models which represent emotions in a well-handled way by machines: discrete emotional model and dimensional emotional model.

The Discrete emotional theory assumes “big six” emotions as basic emotions and their combinations to compose other emotions. Most of the existing SER systems focus on the basic emotional categories often added by a “neutral” emotional state.

The Dimensional emotional model is an alternative “continuous” approach, where emotions are not supposed to be independent of each other. The two-dimensional model is the most usual (Figure 1). The vertical axis expresses arousal or activation which represents the strength of the human disposition to act (Anne, Kuchibhotla & Vankayalapati, 2015). It ranges between apathetic and excited. The horizontal axis represents the valence state, that is whether an emotion is positive or negative. For instance, anger can be translated as an emotional state with negative valence and high arousal. However, some emotions like fear and anger become identical. A proposed solution is to add a third axis like power (Kehrein, 2015), which measures how dominant or submissive an emotion’s expression is. Some researchers use “stance” as the third dimension (Figure 2). Two essential examples of this 3D model are fear which is associated with the “flight” response and anger which is related to the action pattern “fight,” and so on (Kim & Andre, 2016).

It is worth reporting that both models have disadvantages. A common one is their difficulty in the definition of some emotional states. The discrete model fails with high complexity emotions and the dimensional model cannot categorize some emotions that lie outside the dimensional space, like the surprise. Lack of intuition and the need for special training in order to label some emotions are some extra disadvantages of the dimensional model (Zeng, Pantic Roisman, & Huang, 2009).

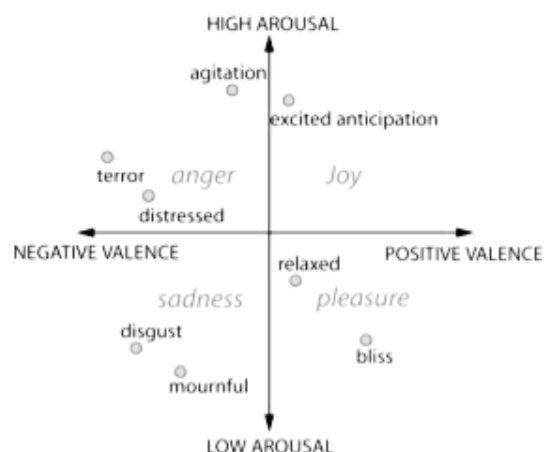


Figure 1. Two-dimensional model by valence and arousal, extracted from Kim & Andre (2016).

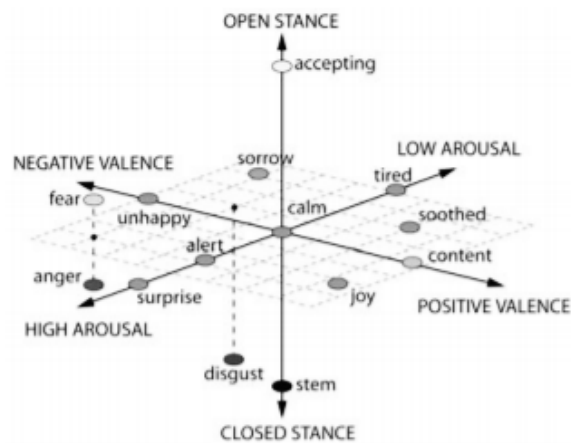


Figure 2. Three-dimensional model for emotion representation by valence, arousal, and stance as it was proposed by Kim & Andre (2016).

1.1 Databases

Once a representation model of emotion is decided upon, the next essential part of speech emotion recognition is the acquisition of labeled data for training and testing that suits the model (Schuller, 2018). Obviously, the quality of the corpus plays a crucial role in the recognition process because of the fact that inappropriate data could lead to incorrect predictions. Currently, there are three categories of databases: Simulated, Elicited, and Natural speech emotion databases. Simulated databases are created by recording actors perform sentences with different emotional content. It is easy to create them but they cannot convey real-life emotions adequately. On the other hand, the Elicited databases are recorded when simulated emotional states are laid in speakers and their emotions are stimulated. They are not fully elicited but close to reality. The last category uses natural speeches from movies, call center recordings, TV or radio shows. Audio data in Natural speech databases, against other databases, carry information as speech uttered in a spontaneously emotional way but it is difficult to collect them due to ethical and legal difficulties. However, we are going through the big data era and it seems feasible to collect more emotional data out of the lab in the next few years.

1.2 Speech Signal Representation

Natural speech is a signal or waveform and is represented by $x(t)$, where time t runs on a continuum. In contrast, time on digital audio signals is discrete. In other words, the digital signal represents samples of natural speech on discrete-time n . This process is known as sampling and leads to vectors or real numbers that depict digital audio signals. To make it more conceivable let's see a simple example, where time 0 is supposed as the time of measurement commenced, that is the time when the first sample is taken. The second sample is taken at 0.002s, the third one at 0.004s, and so on. As we can observe, the time instances

are equidistant with a stable difference, which is celebrated as the sampling period and is equal to $T_s=0.002s$. The inverse of T_s is known as sampling frequency and is defined as $F_s = \frac{1}{T_s}$ Hz. In this case, $F_s = \frac{1}{T_s} = \frac{1}{0.002} = 500$ Hz means that 500 samples of the natural speech signal are taken every second.

Another metric that is useful in the representation of audio signals in the computer language is the sample's depth. The number of bits used to represent each sample and can be equal to 8, 16, 24 or 32 bits. In addition, the audio signals are monophonic (MONO) or stereophonic (STEREO), which means that they are represented by a column vector or a matrix with two columns respectively. The first column of a STEREO signal is called left channel and the second column is called right. Properties of audio signals vary over time, making signals more or less stable.

1.3 Short-term audio processing

A common technique in audio analysis and processing is that the input audio signal is divided into overlapping or non-overlapping frames and then a set of features are extracted for each frame. This procedure results in a sequence of feature vectors per audio signal. As Giannakopoulos, Smailis, Perantonis, & Spyropoulos (2014) describe, the processing of the feature sequence on a mid-term basis is used as a second step in audio feature extraction. Specifically, the short-term processing stage is carried out for mid-term segments of the original audio signal. Subsequently, statistics for the feature sequence, which has been extracted from a mid-term segment, are computed. In practice, a typical range of mid-term duration is 1 – 10 secs, depending on the application domain.

1.3.1 Framing and Windowing

Properties of audio signals vary over time, making signals more or less stable. As a result, emotion can change, usually rapidly. For that reason, audio signals are broken into fixed-length segments, namely frames. To reduce the difference between neighboring frames, we can allow an overlap of 30%-50% between them. These frames are known as short-term windows, and the analysis is carried out on a frame basis (Giannakopoulos & Pikrakis, 2014). After the framing procedure, windowing follows. Suppose a speech signal $x(n)$, aparted from N samples and $n = 0, 1, 2, \dots, N-1$. During each processing step, a small part of the signal is processed, through the multiplication of the audio signal with a shifted version of a finite duration window function $w(n)$. The part of the signal that is processed at the i_{th} step is :

$$x_i(n) = x(n) \cdot w(n - m_i), i = 0, 1, \dots, K - 1,$$

where K is the number of frames and m_i is the number of samples by which the window is shifted in order to yield the i_{th} frame. A usual window is the rectangular one, which is described in the following equation :

$$w_n = \begin{cases} 1, & 0 \leq n \leq W_L - 1 \\ 0, & \text{elsewhere} \end{cases}$$

where W_L is the length of the moving window.

This rectangular window is the simplest one, but discontinuities at first and last points of the signals can be created, which in sequence create problems with Fourier analysis. Consequently, an alternative window is used which is known as Hamming window, and helps in keeping the continuity of the edges of the frame (Rao, Koolagudi, Vempada & Reddy, 2013).

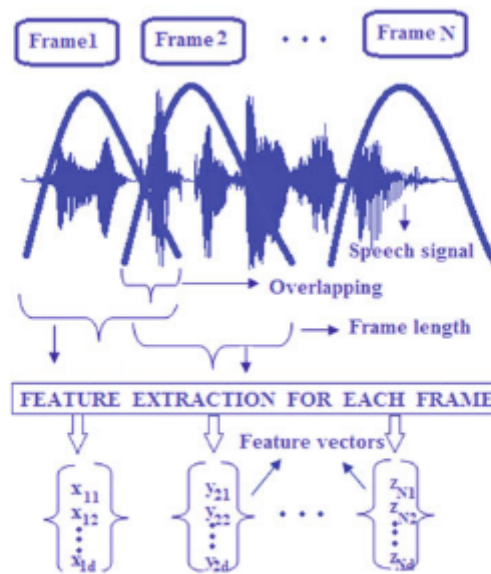


Figure 3. The process of splitting the speech signal into frames, applying a Hamming Window for each frame and its corresponding feature vector. Extracted from Kuchibhotla, Vankayalapati, Vaddi & Anne (2014)

It is worth reporting that the size of each frame must not be too big or too small. Indeed, usually the frame size (in terms of sample points) is equal to the powers of 2, such that it is suitable for Fourier Analysis.

1.3.2 Extraction of Prosodic features

Natural speech includes voiced speech, unvoiced speech, and parts of silence. Voiced speech is produced with the oscillation of vocal folds during the pronunciation of phonemes, where a periodic excitation to the vocal tract is created. Unvoiced speech is generated when the vocal folds are too slack to vibrate periodically and as a result, transient and turbulent noises are produced.

The presence of voiced speech can be detected because of its periodicity. This process is usually called endpoint detection, helps in feature selection and the classification of emotions. Especially prosodic features are those features that can be extracted from the speech during this process. They occur when sounds are put together in a connected speech and are influenced by vocal fold activity (Kuchibhotla et al., 2014). For example, intonation, loudness, pitch, and rhythm are some prosodic features that can easily be perceived by humans and help them classify emotions implicitly. In addition, according to Zeng (2009) and Luengo, Navas, & Hernaez(2010) prosodic features carry essential information of emotional content for a SER and discriminate emotions of high arousal (happy, anger) with these with low arousal (sadness, boredom) respectively. The most frequent prosodic features that are used by researchers are Zero Crossing Rate, Short time Energy and Pitch.

Zero Crossing Rate

Zero-Crossing Rate (ZCR) is a measure of the number of times in a given frame that the amplitude of the speech signals changes from positive to negative and vice versa, divided by the length of the frame (Figure 4). Given the sequence of audio samples of the i_{th} frame $x_i(n)$, $n = 1, 2, \dots, W_L$ where W_L is the length of the frame, the following equation defines the ZCR:

$$Z(i) = \frac{1}{2W_L} \sum_{n=1}^{W_L} |sgn[x_i(n)] - sgn[x_i(n-1)]|$$

where $sgn(\cdot)$ is the sign function, i.e.

$$sgn[x_i(n)] = \begin{cases} 1, & x_i(n) \geq 0 \\ -1, & x_i(n) < 0 \end{cases}$$

It's worth mentioning that researchers suggest that unvoiced speech yields high ZCR, while voiced speech results with low ZCR (Bachu, Kopparthi & Adapa, 2010).

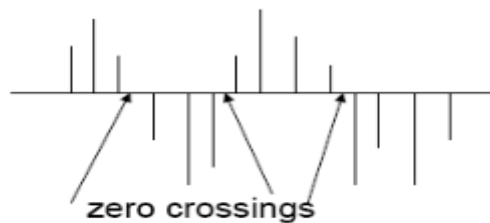


Figure 4: Definition of zero-crossings rate (ZCR)

Short time Energy

The amplitude of the speech signal varies with time. Samples of unvoiced speech of a frame have much lower amplitude than those samples with voiced speech. The energy of the speech signal is a measure that represents the amplitude variations. The following equation defines short-time energy:

$$E(i) = \sum_{n=1}^{W_L} |x_i(n)w|^2$$

Usually, a normalized formula of the energy, the so-called power of the signal, is used in place of the original energy formula. That is:

$$E(i) = \frac{1}{W_L} \sum_{n=1}^{W_L} |x_i(n)w|^2$$

Speech signals contain weak phonemes and short periods of silence between words. As a result, the voiced speech has high energy while low energy is observed in the unvoiced speech. In addition, it has been observed that the energy of high arousal emotions such as anger, happiness, or surprise is high, whereas disgust and sadness yield decreased energy (Lin et al., 2012).

Pitch

Pitch is the fundamental frequency, F_0 , of audio signals, which is related to the highness or lowness of a sound. It is created by the vibrations of the vocal cord and yields rhythmic and tonal characteristics of the speech. A popular technique for estimating fundamental frequency is the autocorrelation function. According to Giannakopoulos et al. (2014), this method shifts the signal and computes the correlation (resemblance) between shifted and original signal, for each signal shift (lag). Finally, the fundamental period is chosen to be the lag, for which the signal best resembles itself, i.e. where the maximum of autocorrelation is.

The autocorrelation function for a frame i is given by the following equation:

$$R_i(m) = \sum_{n=1}^{W_L} x_i(n)x_i(n - m)$$

where W_L is the number of samples per frame and m is the time-lag. In other words, the correlation of the i_{th} frame with itself at time-lag equal to m is signified with $R_i(m)$.

Then the normalized autocorrelation function is calculated:

$$\Gamma_i(m) = \frac{R_i(m)}{\sqrt{\sum_{n=1}^{W_L} x_i(n)^2 \sum_{n=1}^{W_L} x_i(m-n)^2}}$$

The maximum value of $\Gamma_i(m)$ defines the fundamental period:

$$T_0^i = \operatorname{argmax}_{T_{\min} \leq m \leq T_{\max}} \{\Gamma_i(m)\}$$

Then, the fundamental frequency is:

$$f_0^i = \frac{1}{T_0}$$

The change of the fundamental frequency over the course of an utterance produces its fundamental frequency contour whose statistical properties can be used as features. The energy or ZCR contours can be also used as features. Gross statistics such as the mean, maximum, minimum, and range are applied over the sequence of values within an emotion segment. These are the so-called global statistics features. Regression, first and second-order derivatives provide useful information also (Luengo & Navas, 2005).

1.3.3 Spectral features

The vocal tract activity and the spectral characteristics of the voice are also influenced by the emotional state (Luengo et al., 2010). Characteristics of the vocal tract are well reflected in the distribution of the frequency content of sounds (Koolagudi and Rao, 2012b), i.e. of the sound spectrum. The Fourier Transform of a speech frame provides a convenient representation of the spectrum. Widely used audio features that are based on the Fourier transform are known as spectral audio features and will now be described. Mel Frequency Cepstral Coefficients(MFCC), Linear Prediction Cepstral Coefficients (LPCC), formants are some examples of Spectral features. Similar to prosodic features, a windowing method is used to partition speech signals into 20 to 30ms speech segments, which are used to extract spectral features. In addition, similar to prosody statistics, spectral statistics are calculated too.

MelFrequency Cepstral Coefficients (MFCC)

MFCC are the most widely used spectral features (Kuchibhotla et al., 2014) for representing the short term power spectrum of a speech signal. These features are based on the audio perception of humans, who perceive sound in a non-linear way and cannot perceive frequencies over 1KHz. In order to obtain MFCC from a frame several computational steps are executed:

Step 1: Fourier Transform

Fourier transform is applied for the input data using one of various methods like Discrete Fourier Transform (DFT) (Giannakopoulos et al., 2014) or Fast Fourier Transform (FFT) (Anne et al., 2015).

Step 2: Mel-Scaled Filtered Bank and Log Processing

The resulting spectrum is given as input to a mel-scale filter bank that consists of L filters. The filters usually have overlapping triangular frequency responses (Figure 5). The mel-scale (1000 mels correspond to the perception of a sinusoidal tone at 1 KHz, 40 dB above the human auditory threshold) introduces a frequency warping effect endeavoring to comply with the human auditory perception, where perceived pitch increases all the more slower as we go to higher frequencies and neighboring frequencies can be distinguished more easily in the low-frequency region. In other words, the mel scale is a perceptually motivated scale of frequency intervals, which a human perceives as equally spaced. Over the years a number of frequency warping functions have been proposed, e.g.

$$f_w = 1127.01048 \cdot \log\left(\frac{f}{700} + 1\right)$$

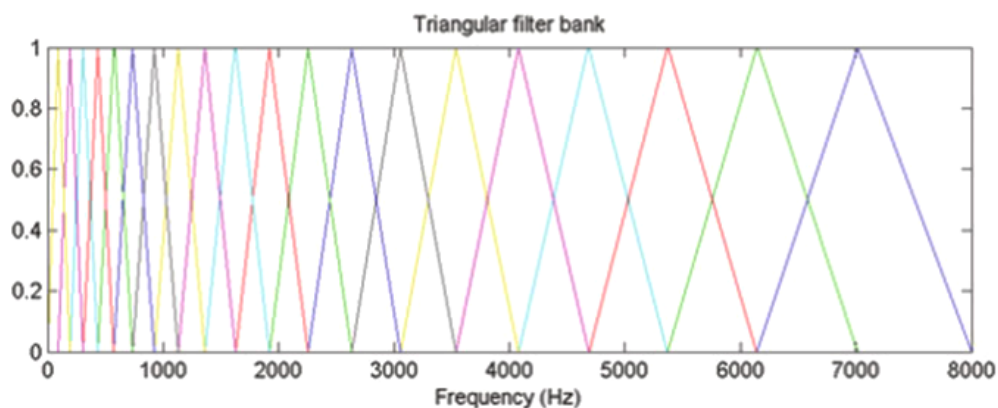


Figure 5. An example of mel-spaced filterbank (Extracted from Anne et al., 2015)

Step 3: Discrete Cosine transform

The final step is the conversion of the log Mel spectrum back to the time domain. Suppose \widetilde{O}_k , $k=1, 2, \dots, L$, is the power at the output of the kth filter, then the resulting MFCCs are given by the following equation:

$$l_m = \sum_{k=1}^L (\log \widetilde{O}_k) \cos \left\{ m \left(k - \frac{1}{2} \right) \frac{\pi}{L} \right\}$$

where $m=1, 2, \dots, L$.

Therefore, MFCCs are the discrete cosine transform coefficients of the mel-scaled log-power spectrum.

Linear Prediction Cepstral Coefficients (LPCC)

Linear Prediction Cepstral Coefficients embodies much emotion-specific information of the speech. Specifically, vocal tract characteristics of speakers can be obtained using Linear Prediction Coefficients (LPC). These characteristics convey differences between emotions and LPC are equivalent to the spectral envelope of the log spectrum of the speech (Wong and Sridharan, 2001), which is determined by the frequency response of the vocal tract and the spectrum of the glottal pulse.

Formant Features

Formants are the frequencies of the acoustic resonance of the vocal tract tube. They depend upon the shape dimension of the vocal tract, showing differences between sounds and as a result between different emotions. The variations of the vocal tract can be represented graphically through visualization of the speech signal's variations of the spectral properties with respect to time. Specifically, the Short-Term Fourier Transform (STFT) can be represented as a matrix with columns apart from the Discrete Fourier Transform (DFT) coefficients of each frame. That is, the column index of this matrix represents time and the row index is associated with the frequency of the respective DFT coefficient. The matrix which is computed from the magnitude of each coefficient can be treated as an image, namely a spectrogram. This image is a representation of the evolution of the signal in the time-frequency domain. In order to generate the spectrogram, we can alternatively use the magnitude of the squared magnitude of the STFT coefficients on a linear or logarithmic scale (dB). The formant frequencies correspond to the dark regions of the spectrogram, which corresponds to energy and can be computed as amplitude peaks in the frequency spectrum of the sound.

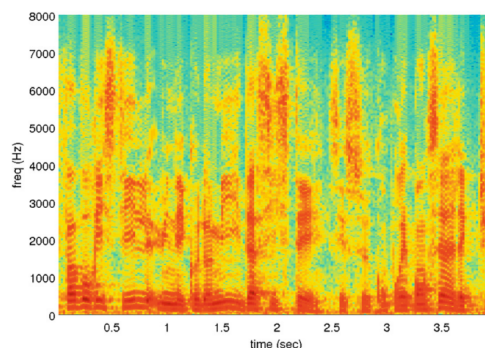


Figure 6. The spectrogram of a speech signal. The frames are non-overlapping, 20 ms long. The horizontal dimension represents time and vertical dimension frequency. Extracted from Giannakopoulos et al. (2014).

1.3.4 Other features

Vocal fold and vocal tract influence prosodic and spectral features respectively, providing useful information of speech (Table 1). These features are the most widely used by researchers and it has been shown that fusion techniques that combine them improves the recognition performance. However, there are additional features that could be extracted and result in classification enhancement.

Emotion	Pitch			Energy	Spectrum
	Mean	Variance	Variation range	Mean	High frequency components
Anger	Highest	Highest	Increase	Highest	Most
Disgust	Lowest	-	Increase	Lowest	Decrease
Fear	Highest	-	Increase	Normal	Increase
Boredom	Lowest	-	Decrease	Lowest	-
Happy	Higher	Increase	Increase	Highest	Increase
Sadness	Lower	Decrease	Decrease	Lower	Decrease
Neutral	Normal	Normal	Normal	Normal	Normal

Table 1. Summary of the effects of several emotion states on selected acoustic features, extracted from Shen, P., & Zhou, C., & Chen, X (2011)

Gobl and Chasaide (2003) highlighted the role of voice quality features, which are determined by the physical properties of the vocal tract and many other studies pointed out a strong correlation between them and the emotional content of the speech. Zhang (2008) and Li, Tao, Chao, Bao, & Liu (2007) added them to the classification process and both achieved about 10% higher recognition rate, compared to the usage of prosodic features alone. Harsh voice, tense voice, modal voice, breathy voice, whisper, creaky voice and lax-creaky voice, jitter, shimmer, and harmonics to noise ratio (HNR) are some voice quality indicators.

In addition, there are features that depend on the Teager Energy Operator (TEO). They are used to detect stress in speech and have been firstly introduced by Teager and Teager (1990) and Kaiser (1990). According to Teager, the muscle tension of a speaker in stressful conditions is affected and results in an alternation of the non-linear flow of air in its vocal tract system that produces sound. Speech is produced by the non-linear airflow in the vocal tract system. Teager and Kaiser developed the Teager Energy Operator (TEO), by considering that hearing is the manner of detection of energy from speech.

1.3.5 Feature selection and dimension reduction

It is essential to observe that there are so many features that could be extracted just from speech signals. In addition, there is no optimal set of features for modeling and classifying

emotional states. If a researcher tries to combine features with a Fusion technique, he will face difficulties because concatenating two feature vectors may result in a feature vector with many dimensions. Once again, the well-known ‘curse of dimensionality’ problem is present. Training time is increased and overfitting occurs that highly affect the prediction rate. As a consequence, feature selection and dimension reduction are important steps for a SER. Random Forest, C4.5 decision trees, Sequential Floating Forward Search (SFFS) are some algorithms, which are used from many approaches and help in the selection of the most significant features of the training data. Ensemble methods are also used like Ensemble Random Forest to Trees (ERFTrees) (Rong et al., 2009). Alternatively, a feature reduction algorithm like principal components analysis (PCA) can be used to encode the main information of the feature space more compactly.

1.4 Related Work

Emotion recognition approaches typically extract features and a classification algorithm is then used to map them to emotion classes. A classification algorithm requires an input X , an output Y , and a function that maps X to Y as in $f(X) = Y$. The learning algorithm uses the labeled data and approximates the mapping function, which helps predict the class of new input.



Figure 7. Extracted from Zhao, Mao, & Chen (2019).

Hidden Markov Model (HMM), Gaussian Mixture Model (GMM), Support Vector machines (SVM), and Artificial Neural Networks (ANN) are usually preferred by researchers. Classifiers based on Decision Trees (DT), k-Nearest Neighbor (k-NN), k-means, and Naive Bayes Classifiers are also often. Lastly, there are SER systems that use ensemble methods that combine several classifiers and obtain better results.

Akçay and Oğuz (2020) summed up classifiers and features that are used in the literature. An obvious inference is that SVMs are the most widely used and successfully applied classifiers, something that was made a breakthrough from (Vogt, André & Wagner, 2008). An example of such an approach is the one of Shen et al. (2011). They achieved 82.5% accuracy using SVM trained on a combination of prosodic and spectral features. It’s worth reporting that in that work the emotional state of happiness has a higher recognition rate than other emotions during all their experiments.

Schuller et al. (2003) compared two methods. In their first method, they derive global statistics features from the raw pitch and energy contour of the speech signal and classify them using GMMs. Each emotion is modeled by one GMM and the maximum likelihood model was considered as the recognized emotion at a time throughout the recognition process. In their second method, the temporal complexity increased by applying continuous

HMM and using low-level instantaneous features rather than global statistics. The average recognition accuracy of seven discrete emotional states exceeded 86% using global statistics, whereas the overall recognition rate on average of five human deciders for the same corpus was 81.3%. The authors mention that some emotions are often confused with certain others and some emotions seem to be recognized again more easily. They accounted for these explaining the difficulties of recorded speakers with feigning certain emotions.

Artificial Neural Networks are also used in emotion recognition systems. Koolagudi and Rao (2012a) used auto-associative neural networks (AANN) to capture the emotion-specific information from excitation source features, GMMs for developing the models using spectral features, and SVMs are used to discriminate the emotions using prosodic features separately. Then, they used fusion techniques in order to combine the three kinds of features, increasing the performance up to 79,14% in the Emo-DB database (Burkhardt, Paeschke, Astrid, Rolfes. Sendlmeier, Weiss, 2005).

Xiao, Dellandrea, Dou, and Chen (2010) used “Dimensional Emotion Classifier” (DEC), a hierarchical two-staged classifier based on the dimensional Emotion model. Unlike most researchers who mainly rely on the classical frequency and energy-based features along with a single global classifier for emotion recognition, they proposed some new harmonic and Zipf-based features to enhance speech emotion recognition. At the first stage, emotional states are classified according to arousal dimension in two sub-stages into three classes: active, median, and passive. At the second stage, the members of these three classes are further classified based on the appraisal dimension. Before each classification stage, feature selection was performed most using the feature selection method SFS (Sequential Forward Selection). They used a back-propagation neural network as a classifier, due to its ability to discriminate between non-linear data and its generalization skills. Berlin and DES databases are used as datasets. For the Berlin dataset, the recognition rate was 68.60% and increased to 71.52% with prior application of gender classification. With the DES dataset 81% recognition rate is obtained.

During the last few years, the performance of the deep learning algorithms exceeded the traditional machine learning algorithms, hence many research efforts have focused on deep learning and the current trend in SER research is no different. Deep learning is a class of machine learning algorithms that uses complex architectures of many interconnected layers, each consisting of nonlinear processing units. Each unit extracts and transforms features. Each layer’s input is the output of the previous one. The advantage of some of these algorithms is that there is no need for hand-crafted features and feature selection. All features are automatically selected and learned from raw data and typically lead to higher performance. Of course, this is achieved with the cost of higher computational time. The most widely used deep learning algorithms and architectures, which have been successfully applied to the emotion recognition domain, are Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN).

Zhao et al. (2019) used two convolutional neural network and long short-term memory (CNN LSTM) networks, 1-D and 2-D CNNs with LSTM network for speech emotion recognition. One 1-D CNN LSTM network was constructed to learn local and global emotion-related features from speech whereas one 2-D CNN LSTM was built to learn

log-mel spectrograms. Both networks share a similar architecture; both have four local feature learning blocks (LFLBs) and one LSTM layer. The LFLB network contains a convolution layer and a pooling layer. On the Berlin Emotional Database (Emo-DB) database, 2-D network obtained validation accuracy of 76.64% and 82.42% with speaker-dependent and speaker-independent experiments respectively. However, validation accuracy is 62.97% and 52.14% on the IEMOCAP database for speaker-dependent and speaker-independent cases, respectively.

Papakostas, Spyrou, Giannakopoulos, Siantikos, Sgouropoulos, Mylonas, and Makedon (2017) presented an approach that uses a Convolutional Neural Network (CNN) as a classifier for emotion recognition. CNNs are responsible for identifying the important features of the input images. In this paper, spectrograms were extracted from raw data and no other extra features were required. Hand-crafted features were only extracted for validation purposes and no linguistic model was required. Their SER is not specific to any particular language and they compared the proposed approach using cross-language datasets. For their experiments four audio datasets were used EMOVO(Constantini, Iaderola, Paoloni & Todisco, 2014), SAVEE (Jackson & Haq, 2011), Emo-DB (German) which are publicly available and a custom made one that includes audio samples gathered from movies. Their results can be seen in Table 2. which indicates that CNNs are able to provide superior results vs. traditional techniques that use hand-crafted features.

		CNN_EM			
		Emovo	Savee	German	Movies
Training Dataset	Emovo	0.57	0.16	0.42	0.27
	Savee	0.30	0.60	0.33	0.31
	German	0.41	0.24	0.67	0.35
	Movies	0.29	0.24	0.42	0.23
Average F1		0.36			

Table 2. Experimental results of Papakostas et al. (2017). Each row indicates the training and each column the testing dataset used in terms of the average F1 measure. Numbers in bold indicate which method achieved the highest performance in each experiment.

Noise in the previous work of Papakostas et. al. (2017) played the role of data augmentation, a widely used technique in Computer Vision. This approach extends the work of Papakostas et al. (2017) using audio data warping augmentation, not only by adding noise but also by shifting, changing pitch, or changing the speed of the acoustic signal, before extracting any spectrogram. EMOVO, SAVEE and Emo-DB databases are used to train a CNN with raw data, in an attempt to affirm that deep learning is indeed able to replace typical approaches which need feature extraction. Cross-language testing is also tried.

Chapter summary

- Emotion representation is based on Discrete emotional theory or the Dimensional emotional model. The first assumes “big six” emotions as basic emotions and their combinations to compose other emotions when the second uses a “continuous” approach, where emotions are not supposed to be independent of each other. However, both models have disadvantages.
- Three categories of databases are used for emotion recognition: Simulated, Elicited, and Natural.
- Audio signals can be represented through vectors or real numbers. This method is known as sampling and helps in speech signal representation.
- Framing is an important step of audio processing, where the input audio signal is divided into overlapping or overlapping frames (short-term windows). Various features are extracted for each frame and a sequence of feature vectors per audio signal is constructed.
- After framing, the windowing procedure follows. The rectangular window is the simplest one, but the Hamming window is the most usual, due to its ability to overcome discontinuity problems.
- Extraction of Prosodic features is essential for emotion recognition. The most frequent prosodic features are Zero Crossing Rate, Short time Energy and Pitch.
- Spectral features of the voice are also influenced by emotions. Mel Frequency Cepstral Coefficients(MFCC), Linear Prediction Cepstral Coefficients (LPCC), formants are some examples of Spectral features. The Fourier transform plays a crucial role in this process.
- Feature selection and dimension reduction are important in order to handle the ‘curse of dimensionality’ problem which arises because of the large number of features that are extracted.
- Work related to emotion recognition that uses classifiers ranging from typical to ensemble and deep learning techniques are presented.

2. Convolutional Neural Networks

During the last decade, deep learning approaches have shown enormous potential in the machine learning field by achieving breakthrough results in Computer Vision and Speech Processing. In literature, many different approaches and deep learning structures have been proposed, but the most dominant seems to be a class of Neural Networks that was firstly reported from LeCun(1989), namely Convolutional Neural Networks (CNNs). CNNs can be considered as an alteration of traditional Neural Networks that combine three architectural ideas: local receptive fields, shared weights, and sometimes, spatial sub-sampling (LeCun , Haffner, Bottou, & Bengio, 1999). A typical Convolutional Network can be seen in Figure7. The input plane receives images and each unit in a layer receives inputs from a set of units located in a small region in the previous layer, without taking into consideration the pixel values of the whole image. In other words, each unit is connected to local receptive fields, an idea inspired by Hubel and Wiesel's (1962) discovery of locally-sensitive, orientation-selective neurons in the cat's visual system. These local receptive fields result in extracting elementary visual features as oriented lines, edges corners or small details in objects' images, even speech spectrograms. These features are then combined by the higher layers in order to detect higher-order features, which are more abstract. The features are extracted in a hierarchical way, an ability that made CNNs popular and emulates the deep and layered learning process of the primate's ventral pathway of the visual cortex in the human brain (Laskar, 2018).

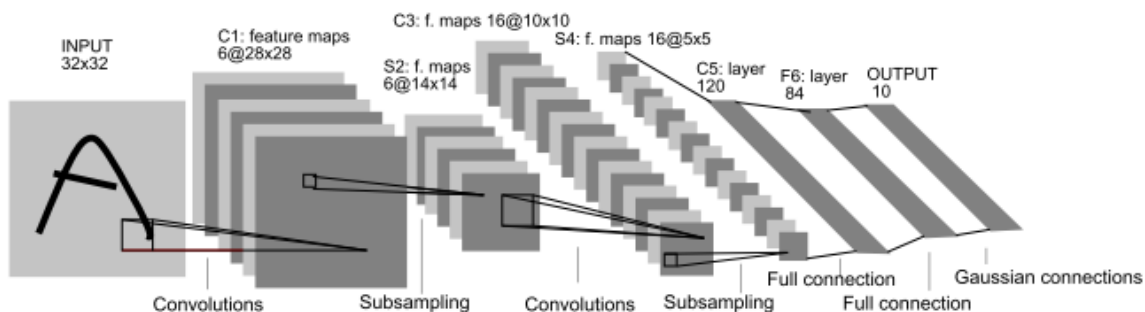


Figure 8. A typical convolutional network for recognizing handwritten digits LeNet-5, from LeCun(1999). Each plane is a feature map, i.e a set of units whose weights are constrained to be identical.

During training, CNN learns through a forward propagation of the data and backpropagation of the error, by updating the weights according to the target. In addition, the most common learning algorithms of such architectures are gradient-based with the most important being the Stochastic-Gradient-Descent and the Adaptive Gradient. In addition, CNNs may consist of an arbitrary number of layers, where each layer may have a different number of nodes and learn spatial hierarchies of patterns (Figure 9). For example, a first convolution layer will learn small local patterns such as edges, a second convolution layer will learn larger patterns made of the features of the first layers, and so on. This allows CNNs to efficiently learn

complex and abstract visual concepts. LeCun’s LeNet-5 convolutional neural network (Figure 8) is organized in layers of two types: convolutional layers and subsampling layers, but contemporary designs of CNNs, which yield state-of-the-art results, conclude layers that can be discriminated into four different categories, namely the (a) Convolutional Layers; (b) Pooling Layers; (c) Normalizing Layers; and the (d) Fully connected Layers.

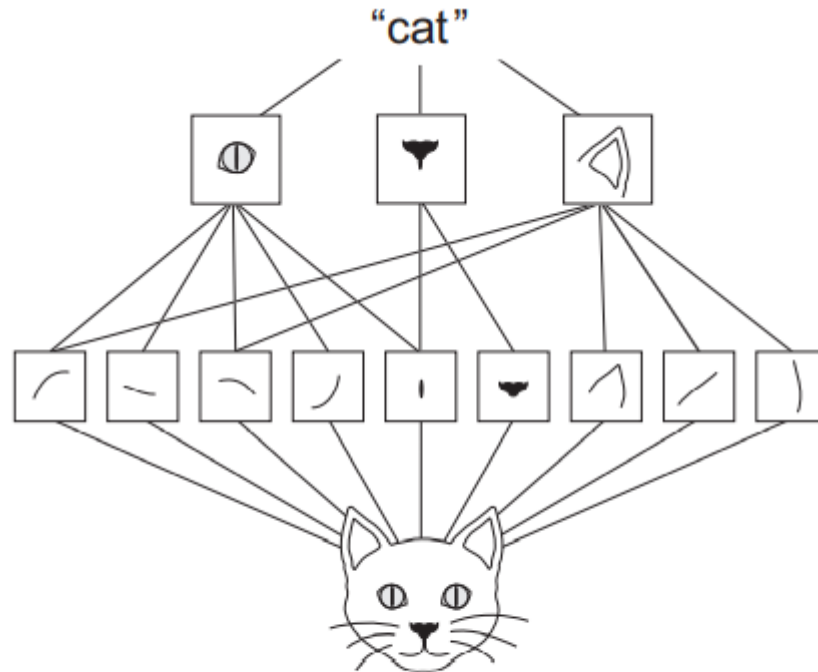


Figure 9. An example of spatial hierarchy. Hyperlocal lines are combined into local objects such as eyes or nose, which then are combined into high-level concepts such as “cat”. Image is extracted from (Chollet, 2017)

2.1 Convolutional layers

A convolutional layer is composed of units. Each unit is a plane, within which, all the units share the same set of weights. The shared set of weights is equivalent to a convolutional kernel (filter) and the set of outputs of the units in such a plane is known as a feature map. Each unit of a feature map performs the same convolutional kernel on different parts of the image and their outputs are then assigned to a non-linear activation function (usually RELU) which results in different patterns of activations for different responses and thus learning of semantic differences in images is facilitated. A complete convolutional layer is composed of several feature maps (with different filters each) so that multiple features can be extracted from each location. The idea of applying the convolution operation to images is not new but the convolution operation in a neural network is an innovation, due to the fact that the values of the filter, the shared weights are trainable. As a consequence, the network will learn what types of features are most useful to extract from the input, for successfully classifying the data, by sliding the kernel on the image. Furthermore, weight sharing helps in reducing the number of free parameters, thereby reducing the “capacity” of the network. For example, the

network of Figure 7 contains 345,308 weights, but only 60,000 of them are trainable, due to weight sharing. Something else that is worth reporting is the annotation of LeCun et al. (1999) that if the input image is shifted, the feature map output will be shifted by the same amount, but will be unchanged otherwise. This interesting property is believed to be the cause of convolutional networks' robustness to input shifts and distortions. Lastly, convolution operation may further be categorized into different types as it is influenced by the type and size of filters, type of padding, size of stride, and the direction of convolution (Chollet, 2017).

2.2 Pooling Layers

Feature patterns that result as patterns of activations can occur at different locations in the image. Once a feature is extracted, its exact location doesn't count anymore. The attention now is paid to its approximate position relative to other features. Let's see an example of digit classification. If we gain the knowledge that the input image has an endpoint of a roughly horizontal line in the upper right area, a corner in the upper left area, and a circular section in the central right area, we can say that the input image is 5. However, the precise positions of these features are irrelevant for classifying the digit and may harm the operation of recognition due to variations of positions between the different instances. A solution for this problem is the so-called sub-sampling layers, which perform a local operation and a sub-sampling, achieving reduction of spatial resolution of the feature map and boost of the invariance of the output to translational shifts and small distortions (Scherer, Müller & Behnke 2010. Particularly, a sub-sampling or pooling layer comprises feature maps, one for each feature map in the previous layer, usually a convolutional layer. Each unit of the pooling layer uses a receptive field and sums up similar information in the corresponding area in the previous layer's corresponding feature map. Different types of pooling operations are available such as max, average, L2, overlapping, spatial pyramid pooling, etc. According to Chollet (2017) max pooling tends to work better than others. Pooling layers also assist in keeping computational cost low and controlling over-fitting.

2.3 Normalizing Layers

Normalization is a method typically used for preparing data before training. Sometimes the dataset contains numerical data that vary in a huge range, this affects the learning process in a negative way. The normalization method provides a uniform scale for numerical values, ensuring there is no information loss. In spite of normalizing the input data, the values that occur after the application of certain activation functions, like RELU, ELU etc. in the hidden layers are unbounded and may vary across a wide scale during the training process, contrary to sigmoid or tanh function for example, where their outputs belong to the intervals $[0,1]$ and $[-1,1]$ respectively. This results in instability. Researchers have indicated Normalization Layers as a solution to this problem, highlighting their important role for deep neural networks. Normalization Layers typically used just before the activation function, acting as data normalization and in some cases can lead to minor improvements on the classification results. In some cases, this kind of layers act over local input regions thus, discrimination

ability over their neighbours is increased. Two powerful and highly used kinds of normalization layers will now be introduced in detail.

2.3.1 Input-Batch-Normalization

Training of Deep Neural Networks encounters a phenomenon known as Internal Covariate Shift. As the parameters of a layer are updated during training, the distribution of the next hidden layer's inputs changes. This slows down the convergence by forcing learning rates to lower and careful parameter initialization is required. Vanishing Gradient is another problem, where certain activation functions (such as tanh or sigmoid) get stuck in the saturation region as the length of the network increases. Both problems can often lead the network to over-fit or unable to achieve learning and can be solved by normalizing layer inputs over each training mini-batch. This process is called Batch Normalization and today is used in almost all CNN architectures.

Batch Normalization transforms and scales every input dimension before it goes through the non-linearity. Particularly, let suppose a layer with d-dimensional input :

$$x = (x^{(1)} \dots x^{(d)}).$$

and a mini-batch B of size m. The normalization is applied to each activation independently and as a result, it is enough to study what happens with just an activation $x^{(k)}$ to understand the method. For clarity reasons k will be omitted. There are m values of activation x:

$$B = \{x_{1\dots m}\}$$

The Batch-Normalization Algorithm as firstly proposed by Ioffe and Szegedy (2015) is as follows:

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1\dots m}\}$;	
Parameters to be learned: γ, β	
Output: $\{y_i = \text{BN}_{\gamma, \beta}(x_i)\}$	
$\mu_{\mathcal{B}} \leftarrow \frac{1}{m} \sum_{i=1}^m x_i$	// mini-batch mean
$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_{\mathcal{B}})^2$	// mini-batch variance
$\hat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}}$	// normalize
$y_i \leftarrow \gamma \hat{x}_i + \beta \equiv \text{BN}_{\gamma, \beta}(x_i)$	// scale and shift

Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

The x to y transformation is called Batch Normalization Transform and it is referred as $BN_{\gamma,\beta}$ with γ, β be learnable parameters. In addition, ϵ is added for numerical stability and avoiding division with zero. Batch-Normalization reduces internal covariant shift sufficiently and sets feature-map values to zero mean and unit variance, unifying their distribution (Ioffe and Szegedy 2015). Lastly, the flow of gradient is smoothed, acting as a regulating factor, thus the generalization of the network is enhanced.

2.3.2 Local Response Normalization Layers

Local Response Normalization (LRN) was first introduced by Krizhevsky, Sutskever & Hinton (2012). They introduced AlexNet Deep Neural Network and used RELU as activation function, opposed to the more common *tanh* and *sigmoid* at that time, and proposed Local Response Normalization in order to help generalization. As they mention, their normalization scheme implements a concept in Neurobiology that is called lateral inhibition and refers to the capacity of a neuron to reduce the activity of its neighbors (Krizhevsky, Sutskever & Hinton, 2012). This process yields local contrast enhancement so that next layers use the locally maximum pixel values as excitation.

Let's express the activity of a neuron by applying a filter i at (x,y) position before normalization by $a_{x,y}^i$ and after normalization by $b_{x,y}^i$. It is :

$$b_{x,y}^i = a_{x,y}^i / (k + \alpha \sum_{j=\max(0,i-n/2)}^{\min(N-1,i+n/2)} (a_{x,y}^j)^2)^\beta$$

where the sum runs over n “adjacent” units (kernel maps) in the feature map at the same spatial position, and N is the total number of filters in the layer. The ordering of the kernel maps is arbitrary and has to be determined before training. The constants $k, n, \alpha,$ and β are hyper-parameters that need tuning. Krizhevsky, Sutskever & Hinton (2012) used $k = 2, n = 5, \alpha = 10^{-4},$ and $\beta = 0.75,$ achieving a decrease of test error rate of about 2%.

2.4 Fully-Connected Layers

Fully-Connected Layers are the top-level layers of every CNN architecture and are responsible for the classification. A fully connected layer connects all neurons in the previous layer to each of its single neurons and becomes not spatially located anymore. So there can be no convolutional layers after a fully connected layer. However, two or more fully connected layers can be stacked, with the last fully connected layer attached to a loss function, which is used to estimate the final classification error and is responsible for updating the network weights during the back-propagation.

2.5 Dropout

Large deep neural networks are very powerful machine learning systems, but they have to deal with overfitting and they are also slow to use. Dropout is a technique that addresses this problem. The key idea is to randomly drop units out, that is temporarily removing them from the network, along with all their incoming and outgoing connections (Figure 10). Only the reduced network is trained on the data in that stage and the removed units are reinserted into the network in the next stage with their original weights. This prevents units from co-adapting too much. In the simplest case, each unit is kept with a fixed probability p . Hyperparameter p equal to 0.5 seems as the optimal choice for a wide range of networks and tasks (Srivastava, Hinton, Krizhevsky, Sutskever & Salakhutdinov, 2014), although it can be chosen using a validation set. Input units are usually retained with p closer to 1 than to 0.5. Furthermore, when a neural network uses Batch Normalization, dropout is either removed or reduced in strength (Ioffe & Szegedy, 2015).

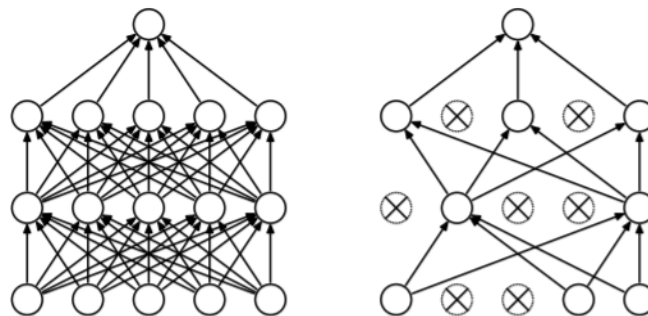


Figure 10. (left): A standard neural network with 2 hidden layers. (right): An example of the network after applying dropout. Crossed units have been dropped. Extracted from (Srivastava, et al., 2014)

During the training of a neural network aparted with n units, dropout samples from 2^n possible different “thinned” networks. At test time, it is easy to approximate the effect of averaging the predictions of all these thinned networks by simply using a single neural net without dropout, which weights are scaled-down versions of the trained weights. If a unit is retained with probability p during training, the outgoing weights of that unit are multiplied by p at test time as shown in Figure 11.

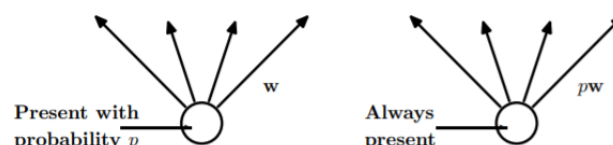


Figure 11. (left): A unit at training time that is present with probability p and is connected to units in the next layer with weights w . (right): At test time, the unit is always present and the weights are multiplied by p . Extracted from (Srivastava, et al., 2014)

It has been shown that dropout reduces overfitting and gives major improvements compared to other regularization methods on supervised learning tasks in vision, speech recognition, document classification, and computational biology.

2.6 Weight Initialization

Training a deep learning network without a useful weight initialization can lead to a very slow convergence or an inability to converge because of exploding gradients, vanishing gradients, or the dying neuron problem. The most widely used weight initialization techniques according to Li, Krček, & Peri (2020) are as follows and a comparison between them is available at Table 3.

All-zeros initialization and Constant initialization

All weights are initialized to 0 (respectively to constant). Also, all activations in all neurons are the same, and therefore all calculations are the same and neurons will learn the same features in each iteration. In Particular, the loss function's derivative is the same for every weight in a weight matrix of a layer. When the values of all the weights are equal, in all iterations, hidden layers become symmetric and the model behaves like a linear one. This problem results in producing poor results not only for zero but also for any constant initialization.

Random initialization

All weight matrix values are initialized randomly close to zero, usually from a normal or a uniform distribution. This technique is generally used to break the symmetry and provides much better accuracy than all-zeros initialization.

For deep networks, the weights can be initialized using heuristics which help mitigate the exploding/vanishing gradients issue. The normal distribution variance is set to k/n_j , where k is a constant depending on the activation function, and n_j is the size of layer j (the number of connections feeding into the node). LeCun (LeCun, Bottou, Orr, Muller, 1998), Glorot/Xavier (Glorot & Bengio, 2016), and He (He, Zhang, Ren & Sun, 2015) have proposed initializers with explained heuristics.

LeCun initialization

- LeCun normal: truncated normal distribution centered on 0 with

$$STD = \sqrt{\frac{1}{n_j}}$$

- LeCun uniform: uniform distribution within [-limit, limit] where the limit is

$$\sqrt{\frac{3}{n_j}}$$

Xavier initialization (Glorot initialization)

- glorot normal: truncated normal distribution centered on 0 with

$$STD = \sqrt{\frac{2}{n_j + n_{j+1}}}$$

- glorot uniform: uniform distribution within [-limit, limit] where the limit is

$$\sqrt{\frac{6}{n_j + n_{j+1}}}$$

He initialization (Kaiming initialization)

- He normal: truncated normal distribution centered on 0 with

$$STD = \sqrt{\frac{2}{n_j}}$$

- He uniform: uniform distribution within [-limit, limit] where the limit is

$$\sqrt{\frac{6}{n_j}}$$

Initialization method	Pros.	Cons.
All-zeros / constant	Simplicity	Symmetry problem leading neurons to learn the same features
Random	Improves the symmetry-breaking process	<ul style="list-style-type: none"> - A saturation may occur leading to a vanishing gradient - The slope or gradient is small, which can cause the gradient descent to be slow
LeCun	Solves growing variance and gradient problems	<ul style="list-style-type: none"> - Not useful in constant-width networks - Takes into account the forward propagation of the input signal - This method is not useful when the activation function is non-differentiable
Xavier	Decreases the probability of the gradient vanishing/exploding problem	<ul style="list-style-type: none"> - This method is not useful when the activation function is non-differentiable - Dying neuron problem during the training
He	Solves dying neuron problems	<ul style="list-style-type: none"> - This method is not useful for layers with differentiable activation function such as ReLU or LeakyReLU

Table 3. A comparison between the main weight initialization techniques according to their advantages and limitations. Extracted from Boulila, Driss, Al-Sarem, Saeed & Krichen, (2021).

Chapter summary

- Convolution neural networks have achieved breakthrough results in Computer Vision as the features are extracted in a hierarchical way from raw data and no feature extraction is needed before training.
- A complete convolutional layer is composed of several feature maps, where each of them correspond to different filter, so that multiple features can be extracted from each location.
- Sub-sampling layers achieve a reduction of spatial resolution of the feature map and boost the invariance of the output to translational shifts and small distortions.
- Batch Normalization transforms and scales every input dimension before it goes through the non-linearity, handling overfitting, Internal Covariate Shift and Vanishing gradients problem.
- Model generalization is boosted with Local Response Normalization Layers.
- Fully-Connected Layers connect all neurons in the previous layer to each of its single neurons. They are the top-level layers of every CNN architecture and are responsible for the classification.
- Dropout that adequately addresses overfitting through randomly dropping units out from the network, along with all their incoming and outgoing connections.
- Weight initialization has been shown to be very useful for the convergence of the network. All-zeros, Random, LeCun, Xavier and He are the most known weight initialization techniques.

3. Method and Experiments

In this work, a SER based on deep learning is implemented. For recognizing five-target emotion labels, a CNN classifier is utilized, which operates upon spectrogram images. Its deep architecture was adapted from Papakostas et al. (2017) and finalized after a very extensive hyperparameter tuning. Our goal was to build a model that could depict robust feature representations for recognizing emotion from speech, extend reference work, and outperform related works. Audio data augmentation techniques such as adding noise, shifting, changing pitch, and changing pitch were applied and the performance was enhanced. The language independence of our system was also tested.

In this section, the emotional speech dataset EMOVO was exhaustively delineated in order to understand in a better way the acted emotion datasets. Afterward, audio preprocessing and the audio augmentation techniques of our approach are stated in detail. Inputs to the network and model's architecture are also presented. Subsequently, Berlin Emotional Database (Emo-DB) and SAVEE databases are used for our experiments, in order to test the robustness of our proposed model and to challenge its language independence. Finally, we present our experimental results and their conclusions. Future work and possible extensions of our approach can be found out in the discussion.

3.1 Emotional speech dataset

For our experiments, we used the EMOVO dataset. This database is the first corpora for emotion recognition in the Italian language. Fourteen sentences (assertive, interrogative, lists) were performed from six actors, three men, and three women. Sentences are based on the big six emotional states (disgust, fear, anger, joy, surprise, sadness) plus the neutral state. The semantic value of the phrases was emotionally neutral in order to abet the actors not to be biased when trying to express the right emotional state. Furthermore, for spectral analysis, some basic linguistic conditions have been satisfied. The recordings were performed with a sampling frequency of 48 kHz, 16-bit stereo, wav format. It is worth noting that only recordings with anger, fear, happiness, sadness, and neutral emotional state were used in this work due to the intention of our results to be comparable to the results of Papakostas et al.(2017). To sum up, 420 samples (14 phrases x 6 actors x 5 emotional states) were used to train our models, the framework of which can be seen in Figure 12.

3.2 Data preprocessing and Augmentation

Before training, audio samples are randomly chopped from the original audio signal in order to have the same duration. We decided to use the same fixed duration as Papakostas et al. (2017), which is 2s. As it is mentioned from theory and is affirmed by many researchers, deep learning requires large amounts of training data, in order to overcome one of the major drawbacks of this approach, namely overfitting. As a result, 420 samples are not enough to achieve satisfactory classification performances. Data Augmentation is a widely used

technique that comes up overfitting from the root of the problem, the training dataset. This is done under the assumption that augmentation could enhance the extracted information from the original dataset and is defined as an artificial increase in the size of the original training samples, through data warping or oversampling. Data warping augmentations transform existing data such that their labels are preserved. In most computer vision approaches that utilize deep learning for classification, data warping encompasses augmentations like adding some noise, horizontally flipping, random crops, or color transformations. Oversampling is the technique that creates synthetic instances and adds them to the original training set. We will not use the oversampling approach in this work.

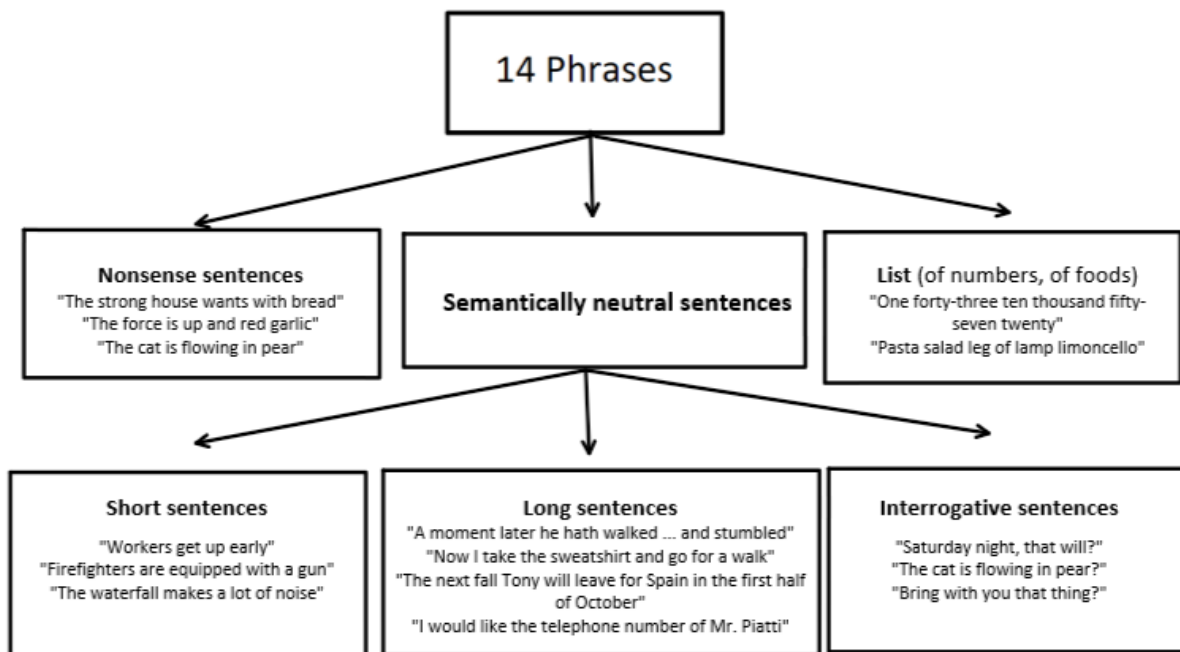


Figure 12. Phrases are translated into English according to (Constantiini et al., 2014)

Noise

In computer audio problems data warping requires different transformations. A classic augmentation is adding noise, by adding a background sound in an exact Signal-To-Noise ratio (SNR). This is achieved as follows. Let suppose $x(n)$ is the initial signal, a "standard normal" distributed random value of *noise*, and a noise *factor*.

The signal with noise is defined : $\gamma(n) = x(n) + noise \cdot \text{sqrt}(factor)$

Then we have :

$$\begin{aligned}
SNR &= 10 \log_{10} \frac{\sum_{n=0}^{N-1} x^2(n)}{\sum_{n=0}^{N-1} (x(n) - \gamma(n))^2} \Leftrightarrow \frac{SNR}{10} = \log_{10} \frac{\sum_{n=0}^{N-1} x^2(n)}{\sum_{n=0}^{N-1} (x(n) - \gamma(n))^2} \\
\Leftrightarrow 10^{\frac{SNR}{10}} &= \frac{\sum_{n=0}^{N-1} x^2(n)}{\sum_{n=0}^{N-1} (x(n) - \gamma(n))^2} \Leftrightarrow 10^{\frac{SNR}{10}} = \frac{\sum_{n=0}^{N-1} x^2(n)}{\sum_{n=0}^{N-1} (\text{noise} \cdot \sqrt{\text{factor}})^2} \\
\Leftrightarrow 10^{\frac{SNR}{10}} &= \frac{\sum_{n=0}^{N-1} x^2(n)}{\sum_{n=0}^{N-1} [\text{noise}]^2 \cdot \text{factor}} \Leftrightarrow 10^{\frac{SNR}{10}} = \frac{\sum_{n=0}^{N-1} x^2(n)}{\text{factor} \sum_{n=0}^{N-1} (\text{noise})^2} \\
\Leftrightarrow \text{factor} &= \frac{\sum_{n=0}^{N-1} x^2(n)}{10^{\frac{SNR}{10}} \sum_{n=0}^{N-1} (\text{noise})^2}
\end{aligned}$$

In our reference paper of Papakostas et al.(2017) a background sound was added in three different Signal-to-Noise ratios (3,4 and 5). The original training set also was included and a 3x dataset increase was achieved. In this approach, some additional data warping techniques for audio samples are used.

Shifting Time

This augmentation shifts the audio signal to left/right with a random second. If the audio signal is shifted to the left for x seconds, the first x seconds will be assigned to zero. On the other hand, if the audio signal is shifted to the right for x seconds, the last x seconds will be assigned to zero.

Change Pitch

This audio deformation is performed by the librosa function. The Pitch of the audio signals is changed randomly.

Change Speed

This audio deformation is a wrapper of the librosa function too and stretches time series by a fixed rate. The new audio signals are augmented by changing the speed of the speaker's voice.

Augmentation factors or parameters were appropriately chosen in order to make subtle but perceptible changes to initial audio signals. An audio signal example and its changes because of different kinds of augmentation can be seen in Figure 13. In our case, we extended the initial training set of size 420 by 420 new samples with random SNR ratio, 420 new samples with random shifting (right or left), 420 new samples with the random change of pitch, and 420 new samples with the random change of speed. The new augmented training set consists of $5 \times 420 = 2100$ audio segments of length 2s each.

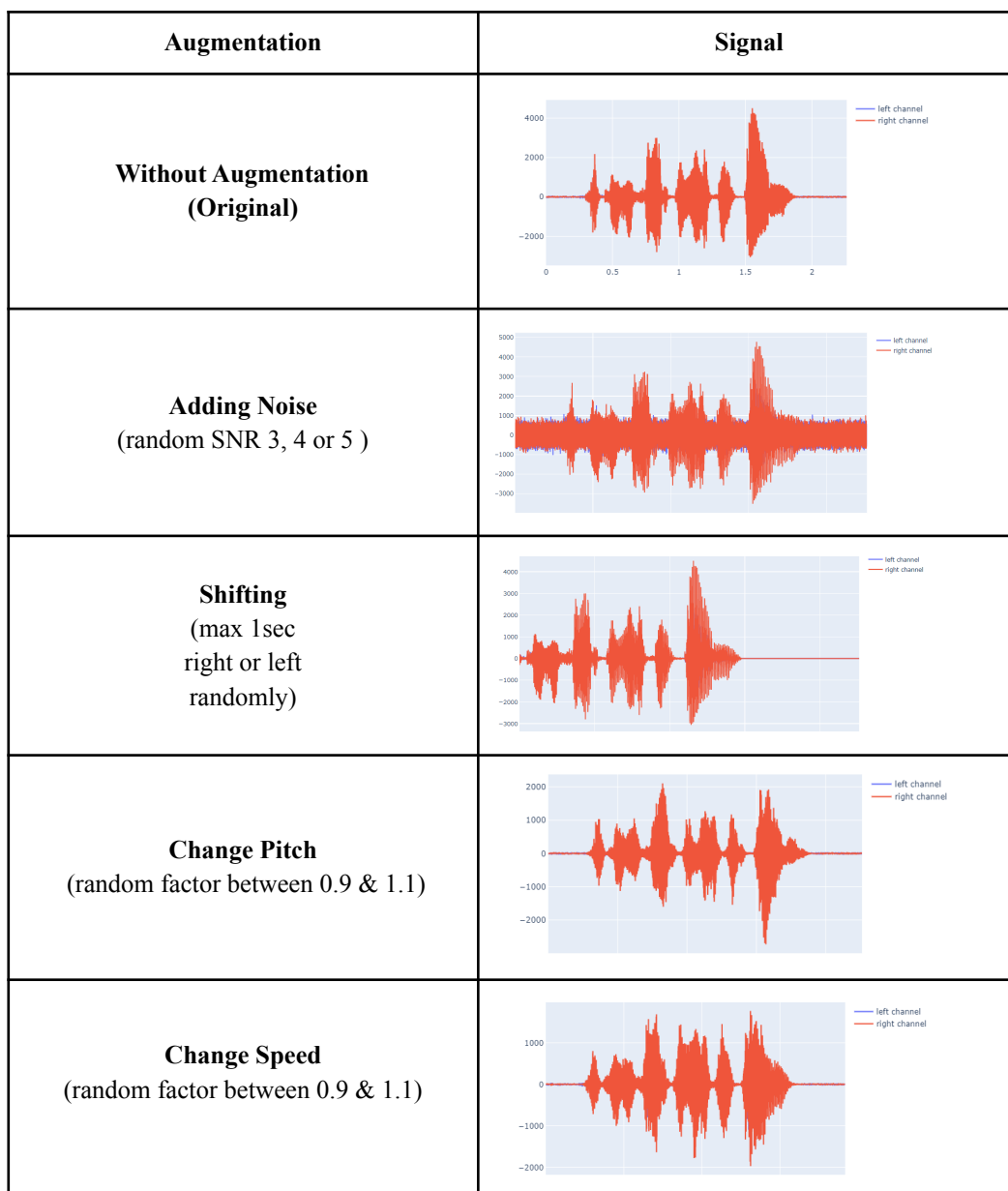


Figure 13. A certain audio signal before and after transformations

3.3 Inputs to network

After data augmentation spectrograms with 40ms short-term window size and 20 ms step are extracted, one for each audio sample. In order to have inputs of the same size to our network, spectrograms were resized to images of 250 x 250. Figure 14 presents an example of five original signals, one for every emotional state, that were augmented with the three different SNR ratios and their spectrograms were extracted. In Figure 15 happiness example spectrograms are enlarged in order to make visible the changes because of noise addition.

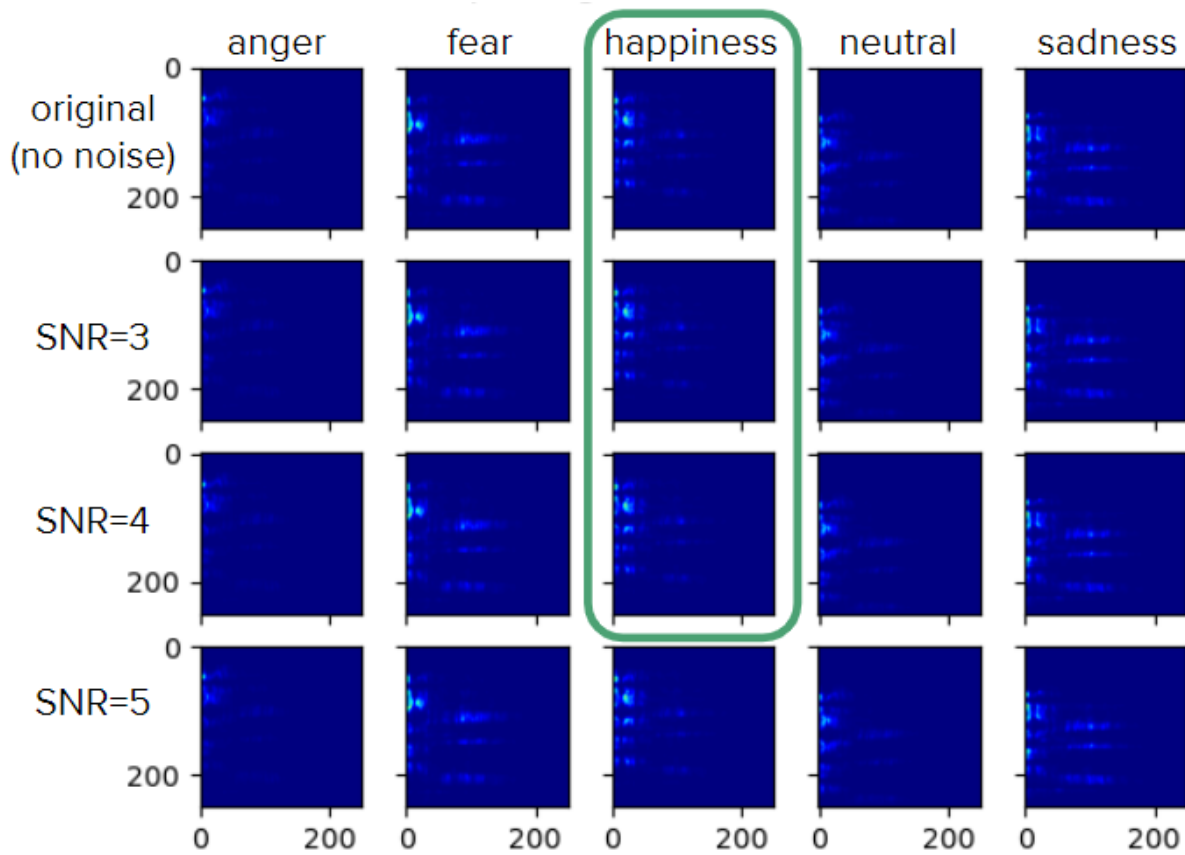


Figure 14. Spectrograms of the same utterance, recorded from the same actor with different emotional content. New spectrograms by adding background noise at three different levels are shown as they can be generated by the augmentation process.

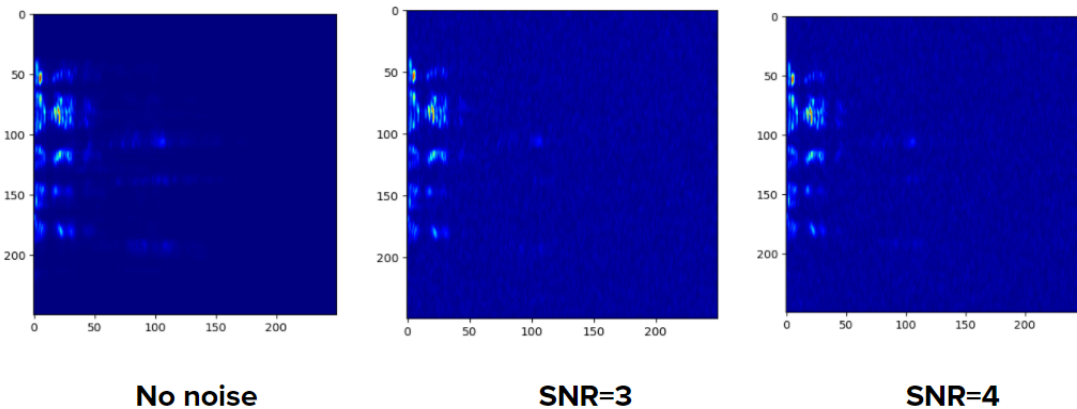


Figure 15. Spectrograms of the utterance with “happiness” content by adding background noise at levels SNR=3, SNR=4.

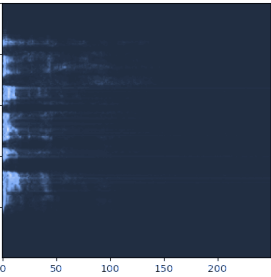
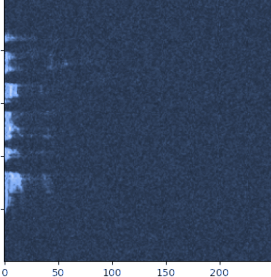
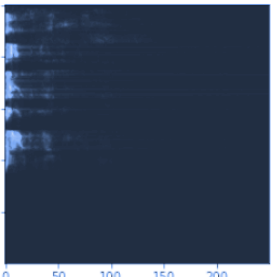
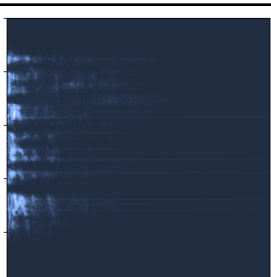
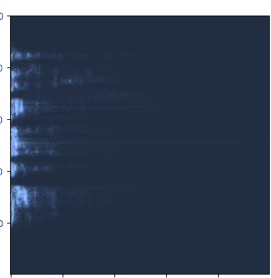
Augmentation	Spectrogram
<p style="text-align: center;">Without Augmentation (Original)</p>	
<p style="text-align: center;">Adding Noise (random SNR 3, 4 or 5)</p>	
<p style="text-align: center;">Shifting (max 1sec right or left randomly)</p>	
<p style="text-align: center;">Changing Pitch (random factor between 0.9 & 1.1)</p>	
<p style="text-align: center;">Changing Speed (random factor between 0.9 & 1.1)</p>	

Figure 16. Examples of part of the augmentation process for an anger sample. The augmentation process generates 4 new spectrograms by adding background noise at a random level, shifting, changing pitch, and changing speed. The colors of spectrograms are transformed for better resolution.

Lastly, in Figure 16 we can observe changes that are derived from all kinds of audio data augmentation for a random sample. In that point, it is important to clarify that spectrograms are images in grey range, but figures represent them pseudocoloured, as matplotlib.pyplot's colormaps.

3.4 Experiments with EMOVO dataset

3.4.1 Network Architecture

For this emotion recognition task, the CNN architecture of Papakostas et al.(2017) was adapted. They chose deep hierarchical visual feature extractors' because of their improved performance compared to classifiers which are trained with hand-crafted features. Our network consists of four stacked blocks interlaced of one convolutional layer with stride 2, a Batch-Normalization layer followed by the non-linearity function, and a max-pooling layer, followed by a Local Response Normalization layer. The first convolutional layer composed of 96 feature maps uses a kernel of size 7. The second and third convolutional layers are composed of 384 filters with a kernel of size 5. The last convolutional layer again uses 384 filters but a kernel of size 3. Batch-Normalization transformation usage before the application of the activation function operates as a normalizer of the input batch. All pooling layers are of stride 2 with their kernel size equal to 3. The last pooling layer is stacked with two fully connected layers of 4096 units with dropout culminating in a softmax classifier. The overall network architecture is illustrated in Figure 17.

For all the layers ReLu is chosen as the activation function and Xavier initialization is adopted for the weights. Standard SGD algorithm is selected for the learning process because of its better results compared to others, using categorical cross-entropy loss function. As a result, the output of the network is distributed on the five target classes. Step decay learning rate scheduler is also adopted, which drops the learning rate every 20 epochs by a factor of 0.1. The initial learning rate is equal to 0.001. In addition, l2 regularizer was added to each convolutional and each fully connected layer with parameter lambda equals to 0.008. These parameters were the result of extensive experimentation and hyperparameter tuning of our model. In Particular, we ran 10-fold Cross-Validation for different combinations of various hyperparameters as can be seen in Table 4 The best performance arose from dropout equal to 0.4 and momentum equal to 0.9. The input to the network corresponds to RGB images of size 250x250 and is organized in batches of 64 samples.

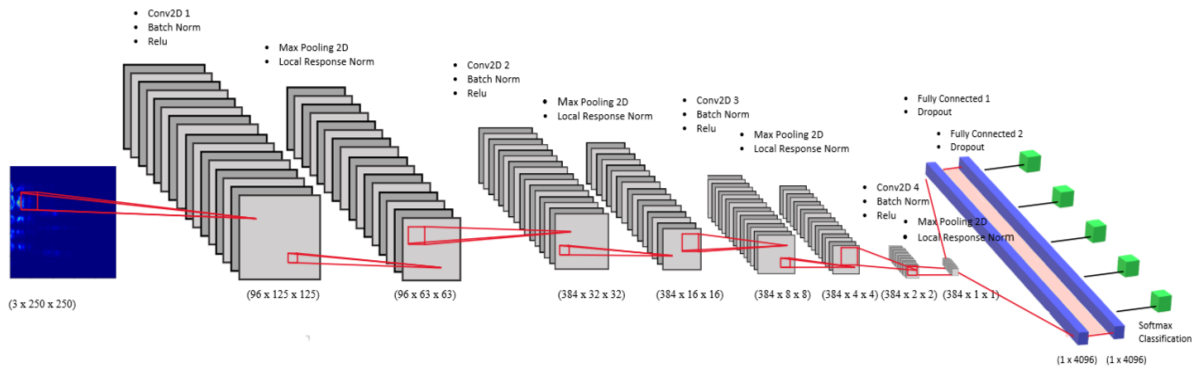


Figure 17. Proposed Convolutional Neural Network Architecture.

Hyperparameters	Values for tuning
l2 weight regularization parameter	0.1 , 0.01, 0.001, 0.005, 0.008, 0.0001, 0.00001, 0.000001
Dropout	0.4, 0.5, 0.6
Initial learning rate	0.1, 0.001, 0.0001
Epochs that learning rate drops	10, 20
Factor of drop	0.1 , 0.2 , 0.4, 0.5

Table 4. Different hyperparameters which were used during the tuning process.

3.4.2 Training Process

The goal of cross-validation is to test the model's ability to predict new data that was not used in the training process, in other words, to give an insight on how generalizable the model will be to an independent dataset. Each round of cross-validation starts with the training of the model on the training set, which is apared from 9 folds. A significant decision is the stopping criterion of the training process. That is the appropriate number of training epochs, for the purpose of preventing overfitting. A simple, effective, and widely used approach is to separate a portion of the training samples into a validation set, which is used for the evaluation of the model after every epoch of the training process. When validation performance stops improving or starts degrading, the training process should be stopped. After that point, the unseen test samples from one fold are used for the evaluation of each round of the 10-fold Cross-Validation. This method is called early stopping and was used in each round.

We first tried to separate our training data before data augmentation and keep 10%

of them as the validation set, as it is classically indicated from the literature. However, because of our belief in the effectiveness of the proposed augmentation technique to produce artificial audio samples in such a good way that could work as the real samples, we tried a validation split of 10% on the augmented data. It seems surprising but choosing the validation set from augmented samples had better performance than the classical approach. As a result, the split that was finally used for hyperparameter tuning and selection of the best model was the split in augmented data. Various models and their behaviors can be seen in Figures 18 and 19.

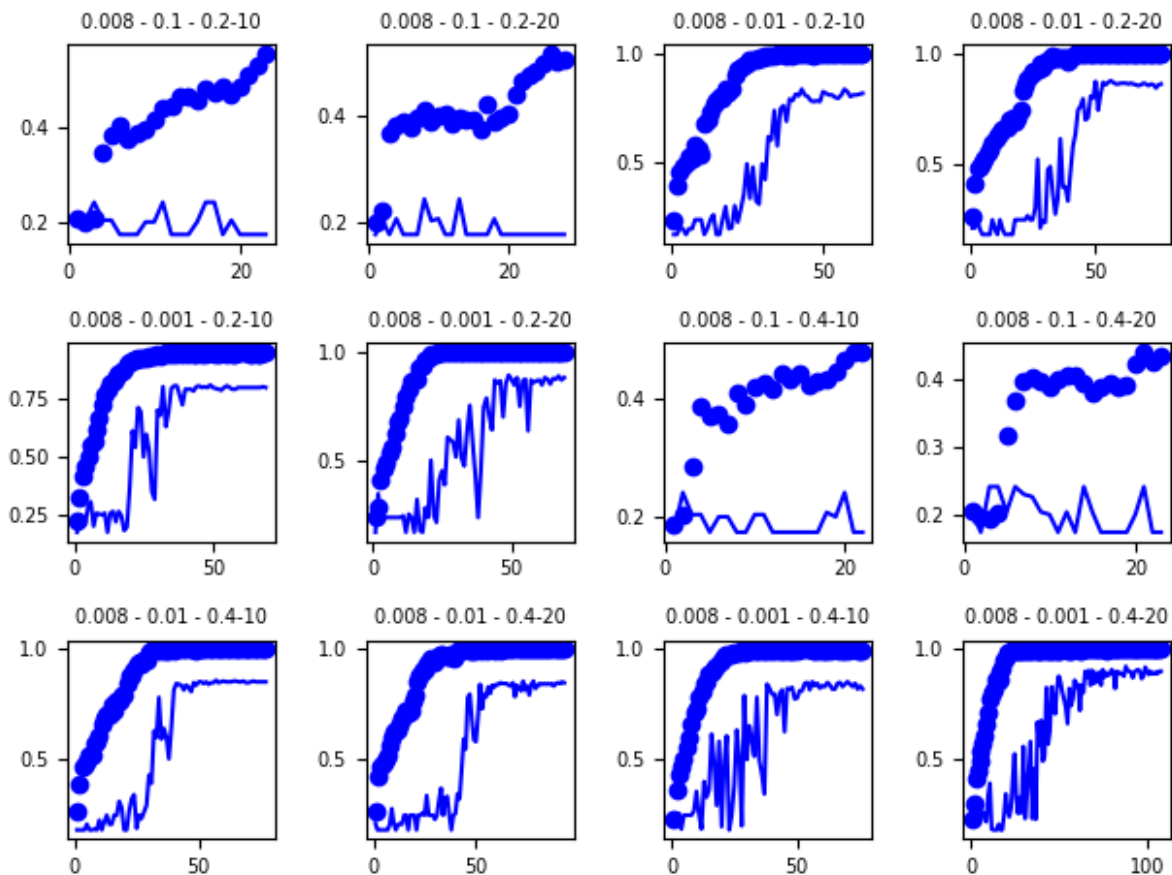


Figure 18 : Learning curves of various models trained on just 80% of training samples and tested with the rest 20%. Title of each curve signifies the combination of the parameters (12 regularization parameter - initial learning rate - factor for learning rate drop - epochs that learning rate drops)

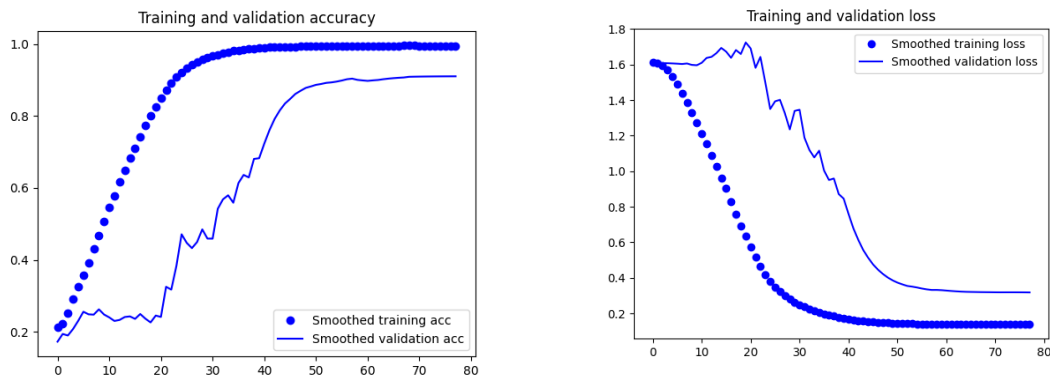


Figure 19: Learning curves of the best model trained just on 80% of training samples and tested with the rest 20%, in order to diagnose the model’s behaviour and performance.

3.4.3 Augmentation Choice

Multiple experiments with different combinations of augmentation combining with different hyperparameters were run in order to find the one with the best performance. Each combination was aparted from original samples plus the augmented samples that were constructed artificially from all original samples (420) and every augmentation. Specifically, we tried augmentations of Table 5, but the performance was optimized with Combination 1 and the model architecture that was previously described.

	Augmentations	Size of final training set (original + augmented samples)
Combination 1	Noise with random SNR , Shifting , Change Pitch , Change Speed	5 x 420 = 2100
Combination 2	Noise with SNR 3 , Noise with SNR 4 , Noise with SNR 5 , Shifting , Change Pitch Change Speed	7 X 420 = 2940
Combination 3	Noise with random SNR ,Shifting , Change Pitch , Change Speed,Shifting +Noise with random SNR , Change Pitch+Noise with random SNR , Change Speed +Noise with random SNR	8 x 420 = 3360
Combination 4	Noise with SNR 3 , Noise with SNR 4, Noise with SNR 5, Shifting, Change Pitch, Change Speed,Shifting +Noise with random SNR , Change Pitch+Noise with random SNR , Change Speed +Noise with random SNR	10 x 420 = 4200

Table 5.

Feature ID	Feature Name	Description
1	Zero Crossing Rate	The rate of sign-changes of the signal during the duration of a particular frame.
2	Energy	The sum of squares of the signal values, normalized by the respective frame length.
3	Entropy of Energy	The entropy of sub-frames' normalized energies. It can be interpreted as a measure of abrupt changes.
4	Spectral Centroid	The center of gravity of the spectrum.
5	Spectral Spread	The second central moment of the spectrum.
6	Spectral Entropy	Entropy of the normalized spectral energies for a set of sub-frames.
7	Spectral Flux	The squared difference between the normalized magnitudes of the spectra of the two successive frames.
8	Spectral Rolloff	The frequency below which 90% of the magnitude distribution of the spectrum is concentrated.
9-21	MFCCs	Mel Frequency Cepstral Coefficients form a cepstral representation where the frequency bands are not linear but distributed according to the mel-scale.
22-33	Chroma Vector	A 12-element representation of the spectral energy where the bins, represent the 12 equal-tempered pitch classes of western-type music (semitone spacing).
34	Chroma Deviation	The standard deviation of the 12 chroma coefficients.

Table 6.

3.4.4 Models

For comparison purposes two extra methods were evaluated :

- **Repetition of Papakostas et al. (2017)** model with augmentation based only on three different Signal-To-Noise ratios (3,4 and 5). Hyperparameters that are interlined on <https://github.com/MikeMpapa/CNNs-Audio-Emotion-Recognition>, code for the reproduction of reference paper's experiments, were used.

- **SVM classification** after hand-crafted features extraction. In particular, mid-term audio feature statistics were extracted and used as inputs to SVM (Table 6).

3.4.5 Results

The next affair that we had to deal with was the evaluation metrics for our emotion classification model. Papakostas et al.(2017) chose average f1 measure, a metric that is widely used. This measure computes the harmonic mean of recall and precision scores in order to summarize model performance into a single metric as follows :

$$f1 - score = 2 \frac{precision \cdot recall}{precision + recall}.$$

Average f1 measure, namely macro-averaged f1, is defined as the arithmetic mean of the per-class f1-scores; it is used when both True positives and True negatives are crucial. It is also considered as a very good metric for imbalanced datasets. For these reasons and with a view to having results that are comparable with our reference paper, f1 metric was chosen as the central metric. However, accuracy, recall and precision were computed during all 10-fold Cross-Validation experiments and are introduced for the best model and the repetition of Papakostas et al (2017) implementation.

Accuracy is the proportion of correctly classified samples. For binary classification problems, recall expresses the proportion of actual positives that are correctly classified and is defined as follows :

$$recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

Respectively, precision expresses the proportion of predicted Positives that is truly positive and is given by the formula :

$$precision = \frac{True\ Positives}{True\ Positives + False\ Positives}.$$

For multiclass classification problems like our emotion recognition task, macro-averaged recall and macro-averaged precision can be computed with respect to the labels. Table 7 presents the experimental results of the best of tried models with our new augmentation in terms of the means of the achieved metrics during 10-fold Cross-Validation, compared to the repetition of Papakostas et al.(2017), SVM for audio features results, and actual f1 performance of our reference paper. A visualization of the comparison between different approaches' f1-scores is also available in Figure 20.

	Accuracy	Precision	Recall	f1
Papakostas et al.(2017)	-	-	-	57.00
SVM for audio features	59.90	-	-	59.70
Repetition of Papakostas et al. (2017)	65.00	58.09	70.00	63.41
CNN with New Augmentation	70.00	65.71	73.99	69.53

Table 7. Experimental results and comparisons to other methodologies.

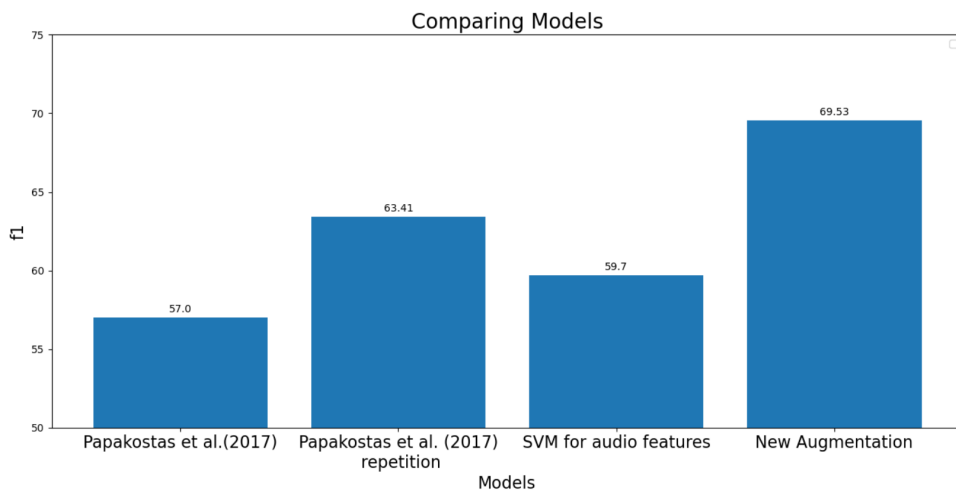


Figure 20. Comparison of the proposed method and comparisons to other methodologies with respect to the average f1-scores.

Obviously, our New Augmentation model achieved about 10% higher performance compared to SNR only Augmentation of Papakostas et al.(2017) and SVM for hand-craft extracted features. The repetition of Papakostas et al.'s (2017) approach results was slightly higher than the original paper. As we can see from this model's box-and-whisker plot (Figure 21) 50 % of the rounds of 10-fold Cross-Validation achieved f1 scores between 58% and 69%, total range is about 22% and positive skewness can be observed. In contrast, the box-and-whisker plot of the New Augmentation model is almost symmetric and closer to the ideal. Furthermore, 50% of its Cross Validation rounds achieved f1 scores between 68% and 72% and the total range is about 8%. These observations show that the New Augmentation model has small variation and the distribution of performances has little dispersion in relation to the median. However, it is significant to report an appreciated outlier round performance with f1-score equal to

81.57%, maybe due to the randomness of splitting data during Cross-Validation. It is a great result, indicating the possibility of the approach to perform better after a more exhaustive tuning of its hyperparameters.

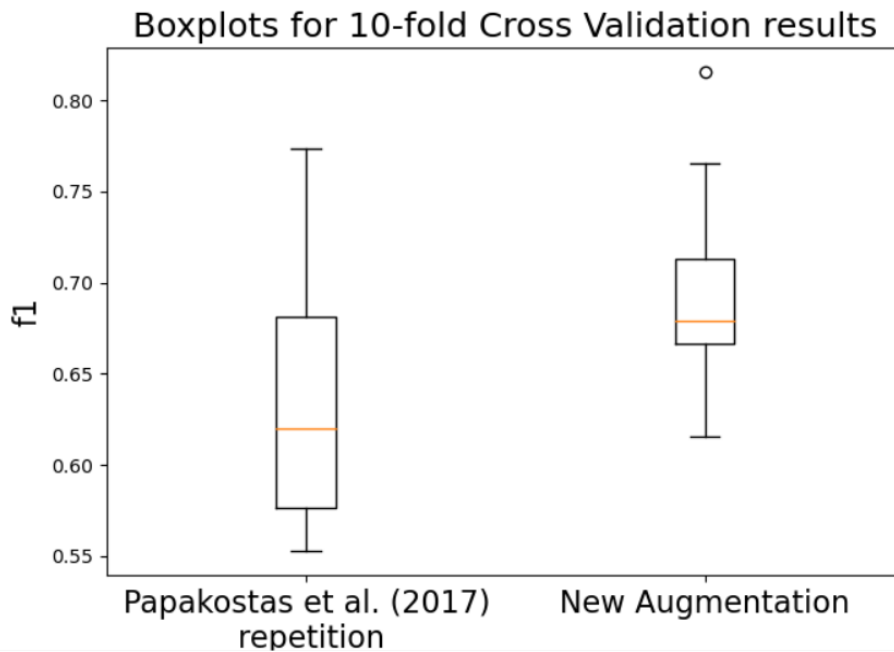
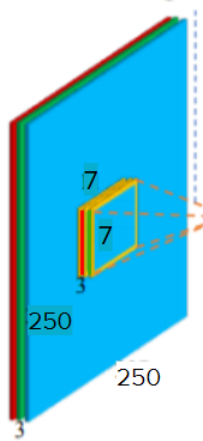


Figure 21. Box-and-whisker plot for f1-scores during 10-fold Cross-Validation of best New Augmentation model and Papakostas et al.(2017) repetition model.

As a consequence, the New Augmentation model can be considered as a robust approach with superiority against other machine learning models. Firstly, Papakostas et al. (2017) represented an approach that uses a CNN functioning as a visual feature extractor, which trained with raw speech information augmented by adding noise and demonstrated that it is able to beat the performance of traditional approaches that use hand-crafted feature extraction. In this work, new augmentation techniques of audio signals enhanced the previous model and resulted in outperformance of its already superior results.

The new augmentation model operates directly on raw data, without the requirement of any linguistic model. Our CNN emotion classifier extracts features based on the final weight values of the layers' filters. These values determine what specific features are detected after the learning process. Figure 23 illustrates the whole filters of the first convolutional layer. Colouring of filters assigns that the darker a region is the higher the impact of the learned weights if on convolution outcome. In contrast, brighter regions correspond to less important learned weights.



All learned filters :
 $3 \times 96 = 288$ filters

Figure 22.

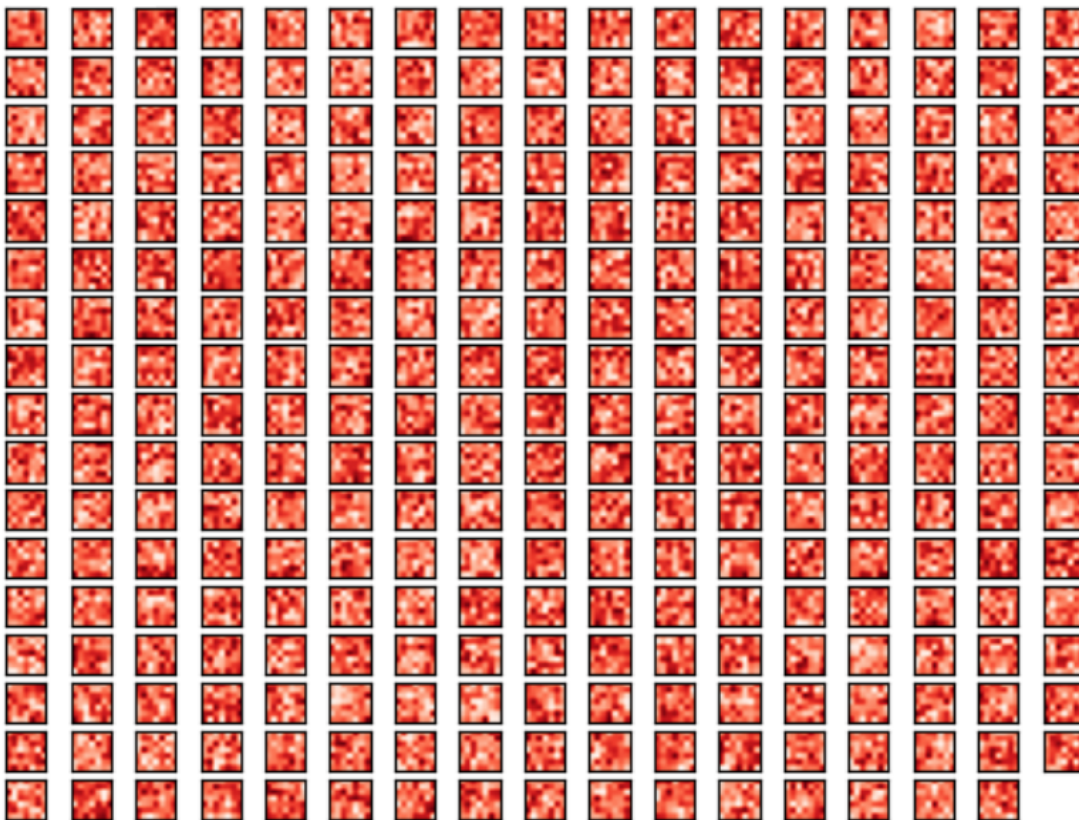


Figure 23. All learned filters of the first convolutional layer.

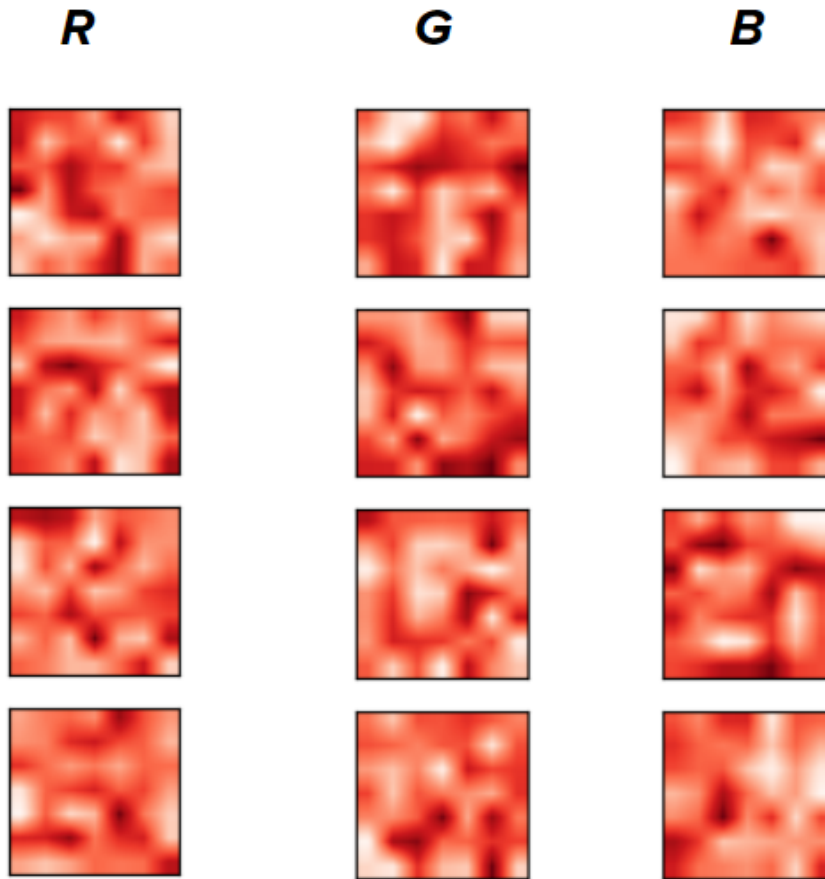


Figure 24. Four randomly selected filters from the first convolutional layer, with their three RGB channels each, as configured after the learning process. Darker regions correspond to the most important learned weights while brighter ones have a lower impact on the convolution outcome.

The inputs of CNN are grey images which are represented with 3 channels (Figure 22), one for each colour (Red - Green - Blue). The filters of the first layer are 96, but each filter has the same number of channels as our input, namely depth. Of course, in our case depth is equal to 3 and for that reason, the first layer’s filters are summed up to 288. In Figure 24, four randomly selected filters from the first convolution layer and their three channels, one for Red, Green, Blue (R-G-B) colours respectively are represented.

3.5 Experiments with other databases and cross-language testing

After the spectacular performance of our proposed model, we wanted to affirm its robustness with other datasets. For this purpose, SAVEE and Emo-DB datasets were used to train and test our CNN, keeping the same values of its hyperparameters. The exact augmentation techniques and the same way of Cross Validation were used in order to build models with only one difference, the training set and as a consequence the language of speech signals. At this point, it's worth reporting that the SAVEE database is composed of records in English

when the Emo-DB dataset is in German. The performance of our SER can be seen at Table 8. As we can see for the EMOVO and Emo-DB dataset our approach outperformed previous work, contrary to SAVEE database where the results are lower. However, the average results remain higher for about 2.66%

	EMOVO	SAVEE	Emo-DB	Average
Papakostas et al. (2017)	57.00	60.00	67.00	61.33
CNN with New Augmentation	69.53	50.63	71.81	63.99

Table 8. Experimental results of CNN with New Augmentation are in terms of the means of the achieved f1 measures during 10-fold Cross-Validation.

Finally, our aim was to check whether our SER depends on language or not. Therefore, we performed cross-language testing. Unfortunately, our approach was not accentuated as a considerably language-independent one with almost all cross-language results being about the baseline, that is 20%.

3.6 Tools and Libraries

The entire code for this project is implemented in the Python Programming Language. For the data augmentation a custom python generator was implemented and the librosa library was used to accomplish changing pitch and changing speed. Spectrograms were extracted, taking advantage of the capabilities of pyAudioAnalysis library (Giannakopoulos, 2015) and for our experiments with CNN models, the KERAS deep-learning framework was used. In addition, we used pyAudioAnalysis to extract mid-term audio feature statistics and run SVM classification. Other python libraries that proved useful were sklearn, matplotlib, pandas, numpy and scipy.

Keras library (Chollet & others, 2015) enables the usage of early stopping with the same name callback through calling it into the fit function. EarlyStopping callback monitors a chosen performance measure for not an improvement. After some preselected number of epochs of not improvement, namely patience, the training process stops and the model with the best weights can be saved. In our case, the performance measure for monitoring was the accuracy of the validation set. A patience of 20 epochs was chosen because it is considered that 15-20 epochs acted as a threshold. That is, the validation accuracy lied below the baseline of 20% by 15-20 epochs and as a result, training stopped early with smaller values of patience, but when the training process exceeded this threshold the validation accuracy started to increase rapidly.

3.7 Discussion and future work

During the last years, speech emotion recognition constituted an area of increased interest for researchers. Many applications of voice user interfaces require emotion recognition solely using audio information. Many machine learning approaches have already been used for this task, but most of them extract hand-crafted features. Deep learning has also been used to classify emotions. The paper of Papakostas et al.(2017) used a CNN and showed that it is indeed able to identify the important features of the spectrograms, extracted from raw audio signals. Moreover, it does not require any linguistic model and is not specific to any particular language.

In this work, new augmentation techniques were proposed which resulted in increased performance, compared to the CNN model of our reference paper. On the contrary, we didn't confirm any language independence, as several other studies have shown when testing cross-language in terms of acoustic emotion recognition (Schuller, 2018), such as the work of Bhaykar, Yadav & Rao (2013). This result may be justified by the strong influence of socio-cultural and linguistic features on expressing emotions in different languages. Furthermore, it is commonly accepted that humans recognize emotions not only from speech but also from gestures, posture, or even eye contact. This additional information is very important for the decision of which is the emotional state of our discussant. In addition, when a human is monolingual, he is trained to recognize the emotions of his native language speakers. It is difficult for him to understand the emotional state of other language speakers. Sometimes, there is also a difficulty in emotion recognition due to sarcasm or irony, even if the speakers' speeches are in the same language. As a result, recognition of irony or sarcasm seems to be a challenging task for future work. Another underlying reason for the difficulty of SERs in language and cross-language recognition may be the weakness of most databases for speech emotion recognition, which are simulated or elicited, to express all the essential emotional information into recorded samples.

Our future goals are focused in various directions. Firstly we would like to increase the robustness of the proposed method in the given datasets, by further optimizing the learning process of the CNN. We also believe that Speaker-Dependent and Speaker-Independent experimental setups will produce outperformed results as recent research has shown. The work of Huang, Dong, Mao, & Zhan (2014) or Zhao et al. (2019) are two standout examples. The new experimental setups are defined as follows :

- Speaker-Dependent: where samples from multiple speakers are used for training and testing takes place on different samples which belong to the same set of speakers
- Speaker-Independent: where samples from multiple speakers are used for training and testing takes place on samples that belong to a different set of speakers

In conclusion, another future goal could be to experiment with models oriented in language or cultural information of the speech or with models that use transfer learning, another possible solution to language independence issues.

Chapter summary

- Pre-segmented audio samples randomly cropped from the original audio signals (2sec) of the EMOVO dataset
- Only 5 classes of emotion (anger, fear, happiness, neutral, sadness)
- New Augmentation was proposed for the crop of the original audio sample which increases the original dataset transforming existing data such that their labels are preserved.
- New Augmentation produces artificial samples by adding a background sound (playing the role of noise) in random Signal-To-Noise ratios (3, 4 and 5), shifting time, changing pitch, or changing pitch.
- For each segment, its spectrogram is extracted, using 40 ms short-term window size and 20 ms step and it is fed as input to the emotion classifier.
- CNN architecture that proposed by Papakostas et al (2017) was adopted for training the model after tuning its hyperparameters.
- 10-fold Cross-Validation was used to evaluate the performance.
- Average f1-score used as our basic metric, but accuracy, average-recall and average-precision were also calculated.
- Augmentation which uses only adding noise also was implemented in order to repeat our basic reference's result for comparison purposes.
- Same, SVM classifier trained with handcrafted features.
- Our proposed New Augmentation outperformed other approaches.

References

- Akçay, M. B., & Oğuz, K. (2020). Speech emotion recognition: Emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers. *Speech Communication*, 116(December 2019), 56–76. <https://doi.org/10.1016/j.specom.2019.12.001>
- Anne, K. R., Kuchibhotla, S., & Vankayalapati, H. D. (2015). Acoustic modeling for emotion recognition. *SpringerBriefs in Speech Technology*.
- Bachu, R. G., Kopparthi, S., Adapa, B., & Barkana, B. D. (2010). Advanced Techniques in Computing Sciences and Software Engineering. *Advanced Techniques in Computing Sciences and Software Engineering*, 5–8. <https://doi.org/10.1007/978-90-481-3660-5>
- Bhaykar, M., Yadav, J., & Rao, K.S. (2013). Speaker dependent, speaker independent and cross language emotion recognition from speech using GMM and HMM. *2013 National Conference on Communications (NCC)*, 1-5.
- Boulila, W., Driss, M., Al-Sarem, M., Saeed, F. & Krichen, M. (2021). Weight Initialization Techniques for Deep Learning Algorithms in Remote Sensing: Recent Trends and Future Perspectives.
- Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W. F., & Weiss, B. (2005). Berlin EmoDB: A database of German emotional speech. *Proceedings of InterSpeech*, 1517–1520. Retrieved from https://www.isca-speech.org/archive/archive_papers/interspeech_2005/i05_1517.pdf
- Chollet, F., & others. (2015). Keras. GitHub. Retrieved from <https://github.com/fchollet/keras>
- Chollet, F. (2017). *Deep learning with python*. Manning Publications.
- Costantini, G., Iaderola, I., Paoloni, A., & Todisco, M. (2014). EMOVO Corpus: an Italian Emotional Speech Database. [Data set]. *9th International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavic, Island, 26–31, pp. 3501–3504. 38.
- Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., & Taylor, J. G. (2001). Emotion recognition - *IEEE Signal Processing Magazine*. *IEEE Signal Processing Magazine*, (January).
- Cowie, R., & Cornelius, R. R. (2003). Describing the emotional states that are expressed in speech. *Speech Communication*, 40(1–2), 5–32. [https://doi.org/10.1016/S0167-6393\(02\)00071-7](https://doi.org/10.1016/S0167-6393(02)00071-7)
- Ekman, P., & Oster, H. (1979). OF EMOTIONI.
- Giannakopoulos, T., Smailis, C., Perantonis, S., & Spyropoulos, C. (2014). Realtime depression estimation using mid-term audio features. *CEUR Workshop Proceedings*, 1213, 41–45.
- Giannakopoulos, T., & Pikrakis, A. (2014). Introduction to Audio Analysis: A MATLAB Approach. *Introduction to Audio Analysis: A MATLAB Approach*. <https://doi.org/10.1016/C2012-0-03524-7>
- Gobl, C., & Ní Chasaide, A. (2003). The role of voice quality in communicating emotion, mood and attitude. *Speech Communication*, 40(1–2), 189–212. [https://doi.org/10.1016/S0167-6393\(02\)00082-1](https://doi.org/10.1016/S0167-6393(02)00082-1)

- Glorot, X. & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Journal of Machine Learning Research - Proceedings Track*, 9, 249-256.
- Han, J., Ji, X., Hu, X., Guo, L., & Liu, T. (2015). Arousal recognition using audio-visual features and fMRI-based brain response. *IEEE Transactions on Affective Computing*, 6(4), 337–347. <https://doi.org/10.1109/TAFFC.2015.2411280>
- He, K., Zhang, X., Ren, S. & Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *IEEE International Conference on Computer Vision (ICCV 2015)*. 1502. 10.1109/ICCV.2015.123.
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *32nd International Conference on Machine Learning, ICML 2015*, 1, 448–456.
- Jackson, P. & Haq, S. (2011). Surrey Audio-Visual Expressed Emotion (SAVEE) database. (1.0.0). [Data set]. UK: University of Surrey: Guildford.
- Kaiser, J. (1990). On a simple algorithm to calculate the 'energy' of a signal. *International Conference on Acoustics, Speech, and Signal Processing*, 381-384 vol.1.
- Kehrein, R. (2003). Die prosodie authentischer emotionen. *Sprache Stimme Gehör*, 27(2), 55–61. <https://doi.org/10.1055/s-2003-40251>
- Kim, J., & André, E. (2008). Emotion recognition based on physiological changes in music listening. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(12), 2067–2083. <https://doi.org/10.1109/TPAMI.2008.26>
- Koolagudi, S. G., & Rao, K. S. (2012a). Emotion recognition from speech using source , system , and prosodic features. 265–289. <https://doi.org/10.1007/s10772-012-9139-3>
- Koolagudi, S. G., & Rao, K. S. (2012b). Emotion recognition from speech: A review. *International Journal of Speech Technology*, 15(2), 99–117. <https://doi.org/10.1007/s10772-011-9125-1>
- Kuchibhotla, S., Vankayalapati, H. D., Vaddi, R. S., & Anne, K. R. (2014). A comparative analysis of classifiers in emotion recognition through acoustic features. *International Journal of Speech Technology*, 17(4), 401–408. <https://doi.org/10.1007/s10772-014-9239-3>
- Krizhevsky, A., Sutskever, I., & Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60, 84 - 90.
- Laskar, M. N. U., Giraldo, L. G. S., & Schwartz, O. (2018). Correspondence of Deep Neural Networks and the Brain for Visual Textures. 1–17. <http://arxiv.org/abs/1806.02888>
- LeCun, Y. (1989). Generalization and network design strategies. In R. Pfeifer, Z. Schreter, F. Fogelman, & L. Steels (Eds.), *Connectionism in perspective* Elsevier.
- LeCun Y., Bottou L., Orr G.B., Müller K.R. (1998) Efficient BackProp. In: Orr G.B., Müller K.R. (eds) *Neural Networks: Tricks of the Trade*. Lecture Notes in Computer Science, vol 1524. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-49430-8_2

LeCun, Y., Haffner, P., Bottou, L., & Bengio, Y. (1999). Object Recognition with Gradient-Based Learning. *Shape, Contour and Grouping in Computer Vision*.

Li, Y., Tao, J., Chao, L., Bao, W., & Liu, Y. (2017). CHEAVD: a Chinese natural emotional audio–visual database. *Journal of Ambient Intelligence and Humanized Computing*, 8(6), 913–924. <https://doi.org/10.1007/s12652-016-0406-z>

Lin, J.-C., Wu, C.-H., & Wei, W.-L. (2012). Error Weighted Semi-Coupled Hidden Markov Model for Audio-Visual Emotion Recognition. *IEEE Transactions on Multimedia*, 14(1), 142–156. <https://doi.org/10.1109/TMM.2011.2171334>

Li, H., Krček, M., & Perin, G. (2020). A comparison of weight initializers in deep learning-based side-channel analysis. In J. Zhou, C. M. Ahmed, M. Conti, E. Losiouk, M. H. Au, L. Batina, Z. Li, J. Lin, B. Luo, S. Majumdar, W. Meng, M. Ochoa, S. Picek, G. Portokalidis, C. Wang, & K. Zhang (Eds.), *Applied Cryptography and Network Security Workshops - ACNS 2020 Satellite Workshops, AIBlock, AIHWS, AIoTS, Cloud S and P, SCI, SecMT, and SiMLA, Proceedings* (pp. 126-143). (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Vol. 12418 LNCS). Springer. https://doi.org/10.1007/978-3-030-61638-0_8

Luengo, I., & Navas, E. (2005). Automatic Emotion Recognition using Prosodic Parameters Department of Electronics and Telecommunication University of the Basque Country , Spain. *Power*, 493–496.

Luengo, I., Navas, E., & Hernaez, I. (2010). Feature analysis and evaluation for automatic emotion identification in speech. *IEEE Transactions on Multimedia*, 12(6), 490–501. <https://doi.org/10.1109/TMM.2010.2051872>

Papakostas, M., Spyrou, E., Giannakopoulos, T., Siantikos, G., Sgouropoulos, D., Mylonas, P., & Makedon, F. (2017). Deep visual attributes vs. hand-crafted audio features on Multidomain Speech Emotion recognition. *Computation*, 5(2), 1–15. <https://doi.org/10.3390/computation5020026>

Plutchik, R. (2001). The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American Scientist*, 89(4), 344–350. Retrieved from <http://www.jstor.org/stable/27857503>

Rao, K. S., Koolagudi, S. G., & Vempada, R. R. (2013). Emotion recognition from speech using global and local prosodic features. *International Journal of Speech Technology*, 16(2), 143–160. <https://doi.org/10.1007/s10772-012-9172-2>

Rong, J., Li, G., & Chen, Y. P. P. (2009). Acoustic feature selection for automatic emotion recognition from speech. *Information Processing and Management*, 45(3), 315–328. <https://doi.org/10.1016/j.ipm.2008.09.003>

Schuller, B. W. (2018). Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends. *Communications of the ACM*, 61(5), 90–99. <https://doi.org/10.1145/3129340>

Schuller, B., Rigoll, G., & Lang, M. (2003). Hidden markov model-based speech emotion recognition. 1–4.

Scherer, D., Müller, A.C., & Behnke, S. (2010). Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. *ICANN*.

Shen, P., & Zhou, C., & Chen, X. (2011). Automatic Speech Emotion Recognition using Support Vector Machine. 2. 621-625. [10.1109/EMEIT.2011.6023178](https://doi.org/10.1109/EMEIT.2011.6023178).

Song, T., Zheng, W., Song, P., & Cui, Z. (2020). EEG Emotion Recognition Using Dynamical Graph Convolutional Neural Networks. *IEEE Transactions on Affective Computing*, 11(3), 532–541. <https://doi.org/10.1109/TAFFC.2018.2817622>

Teager H.M., Teager S.M. (1990) Evidence for Nonlinear Sound Production Mechanisms in the Vocal Tract. In: Hardcastle W.J., Marchal A. (eds) *Speech Production and Speech Modelling*. NATO ASI Series (Series D: Behavioural and Social Sciences), vol 55. Springer, Dordrecht. https://doi.org/10.1007/978-94-009-2037-8_10

Vogt, T., André, E., & Wagner, J. (2008). Automatic recognition of emotions from speech: A review of the literature and recommendations for practical realisation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4868 LNCS, 75–91. https://doi.org/10.1007/978-3-540-85099-1_7

Wong, E., & Sridharan, S. (2001). Comparison of linear prediction cepstrum coefficients and mel-frequency cepstrum coefficients for language identification. *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, ISIMP 2001*, 95–98. <https://doi.org/10.1109/isimp.2001.925340>

Xiao, Z., Dellandrea, E., Dou, W., & Chen, L. (2010). Multi-stage classification of emotional speech motivated by a dimensional emotion model. *Multimedia Tools and Applications*, 46(1), 119–145. <https://doi.org/10.1007/s11042-009-0319-3>

Zhang, S. (2008). Emotion recognition in Chinese natural speech by combining prosody and voice quality features. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5264 LNCS(PART 2), 457–464. https://doi.org/10.1007/978-3-540-87734-9_52

Zhao, J., Mao, X., & Chen, L. (2019). Speech emotion recognition using deep 1D & 2D CNN LSTM networks. *Biomedical Signal Processing and Control*, 47, 312–323. <https://doi.org/10.1016/j.bspc.2018.08.035>

Zeng, Z., Pantic, M., Roisman, G. I., & Huang, T. S. (2009). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1), 39–58. <https://doi.org/10.1109/TPAMI.2008.52>