



SVM-BASED EMOTION RECOGNITION FROM SPEECH WITH GTCC AND FREQUENCY FEATURES

UDC (004.934.2:159.9)

Dragan Veljković, Dejan Rančić

University of Niš, Faculty of Electronic Engineering, Department of Computer Science,
Republic of Serbia

ORCID iDs: Dragan Veljković
Dejan Rančić

 <https://orcid.org/0009-0001-4553-1716>
 <https://orcid.org/0000-0002-9445-7700>

Abstract. *When a person is in a certain emotional state, a large number of physiological changes occur in the body. These changes significantly affect the way words are pronounced compared to neutral speech. This means that the configuration of the vocal tract changes depending on the speaker's emotional state. Furthermore, in emotional speech, physiological changes influence certain speech properties, such as speech rate, intensity, and pitch. Successful classification of emotional speech into the appropriate emotion class requires extraction of salient speech features and construction of a feature vector composed of discriminative attributes that facilitate accurate classification. In this study, we use Gammatone Cepstral Coefficients (GTCC) as components of the feature vector for speech emotion recognition. GTCC are a biologically inspired modification of Mel-Frequency Cepstral Coefficients (MFCC). They are based on gammatone filters, which simulate the human auditory system more effectively than the mel-frequency filters used in MFCC. The remainder of the feature vector is composed of spectral characteristics extracted from the speech signal. In our classification model, the components of the feature vector are primarily extracted by performing spectral analysis on short-time frames of the observed speech signal. Feature vectors constitute discriminative representations that facilitate the more effective classification of speech into corresponding emotional categories. Our classifier is based on Support Vector Machines (SVM), with optimized hyper-parameters.*

Key words: *Speech emotion classification, gammatone filter bank, GTCC, SVM.*

Received February 10, 2025 / Accepted September 03, 2025

Corresponding author: Dragan Veljković

University of Niš, Faculty of Electronic Engineering, Department of Computer Science, Aleksandra Medvedeva 4,
18000 Niš, Republic of Serbia

E-mail: draganastek68@gmail.com

1. INTRODUCTION

The vocal tract spectrally shapes the excitation signal generated by the vocal folds and thereby functions as a spectral filter. Changes in the geometric shape of the vocal tract determine which spectral components will be amplified and which will be attenuated. A word spoken by the same speaker is never pronounced in exactly the same way, meaning that the feature vectors of the spoken word are never one hundred percent identical. This variability is largely influenced by differences in the speaker's emotional and health state, various psychophysical conditions, as well as age. The intensity of emotion, for each speaker, represents an individual characteristic of their expressive style for conveying that emotion.

Speech is a non-stationary random signal, which means that its statistical properties change over time. The reason for this is that structures such as the shape of the vocal tract, the position of the tongue, and the shape of the mouth are variable, i.e., the elements that produce speech are themselves non-stationary and inherit variability from the structures that generate them. Since the shape of the vocal tract is also a variable factor, the features describing it are dynamic as well. The variability of the vocal tract shape is conditioned by emotional state, social factors, aging, health condition, psychophysical state, and ultimately by the act of speaking itself.

The most significant features of the vocal tract, according to the majority of authors who have studied its analysis [8][9][10][11], are pitch (fundamental frequency), speech signal energy, MFCC features, and spectral harmonics.

The frequency characteristics of the speech signal largely depend on the shape of the vocal tract. Based on this observation, the feature vector in this work is composed of the frequency characteristics of the speech signal. The remaining part of the feature vector is constructed using Gammatone Cepstral Coefficients (GTCC). These coefficients are derived from a gammatone filter bank based on the Equivalent Rectangular Bandwidth (ERB) scale, a type of filter bank that mimics the way the human auditory system processes sound. GTCCs are used to capture the spectral characteristics of speech signals. The computation of the proposed gammatone cepstral coefficients is performed in a similar manner to MFCC extraction.

In [1], GTCC coefficients demonstrated superior performance in classifying emotional speech in noisy environments. Some studies have shown that Gammatone Cepstral Coefficients (GTCC) can provide equal or even better performance than MFCC due to the improved filter response characteristics [3][4][5]. The implementation of an emotion recognition model should be "robust" against noise factors, since failure to satisfy this condition directly affects classifier accuracy [12][13]. Using a classifier whose feature vector includes GTCC rather than MFCC yields better results when classifying speech utterances recorded in noisy conditions.

In our work we used a traditional SVM classifier, which achieved a respectable classification accuracy. In recent years, research on emotion recognition from speech has predominantly employed neural networks as classifiers to map speech signals into emotional categories. For example, Prabhakar et al. [20] developed a multi-channel CNN-BLSTM architecture with an attention mechanism for speaker-independent SER, taking into account both phase and magnitude spectral features. Phase features were extracted using the Modified Group Delay Function (MODGD) and combined with MFCC features. The IEMOCAP dataset was used for performance evaluation, and the experimental results demonstrated improvements over MFCC-based and other existing unimodal SER approaches.

Hama Saeed [21] proposed a DNN-based SER approach organized into three stages: feature extraction, normalization, and emotion recognition. From the audio signals, MFCC, Mel-spectrogram, chroma and polynomial (poly) features were extracted. SMOTE was used to

augment minority classes and Min–Max scaling for normalization. The DNN model was evaluated on three commonly used languages (German, English, and French) using the EMO-DB, SAVEE and CaFE corpora, achieving strong results: 95% on EMO-DB, 90% on SAVEE and 92% on CaFE.

Alluhaidan et al. [22] proposed a combination of MFCCs with time-domain features (denoted MFCCT). Using CNNs, the hybrid MFCCT features demonstrated remarkable effectiveness, outperforming both MFCC and time-domain features on benchmark datasets such as Emo-DB, SAVEE and RAVDESS, with accuracies of 97%, 93% and 92%, respectively.

The significance of this study lies in the fact that a unique feature vector was created, used for the first time in the literature, consisting of GTCC values combined with selected spectral features. Classification performance based on this feature vector demonstrated a high degree of accuracy in assigning emotional speech to the corresponding emotional class.

In the second chapter, we describe the speech databases used for training and testing our emotional speech classifier. This is followed by a section describing the Gammatone Filter Bank and the methods of extracting emotional speech features that influence the success of emotion classification. These include spectral characteristics of the speech signal transformed into GTCC, along with Δ GTCC and Δ^2 GTCC features. Next, we present the description and computation of additional spectral features that complement the values of our feature vector. The following section briefly outlines the operation of the Support Vector Machine classifier.

The final section presents the practical classification of speech utterance databases into appropriate emotional classes, where the program for spectral feature extraction and emotional speech classification into specific emotional categories was implemented in MATLAB, version R2023b.

2. SPEECH CORPUS

We can confidently state that a certain percentage of the classification accuracy of a speech emotion recognition classifier depends on the quality of the emotional speech database used for training and testing, as well as on its degree of similarity with real-world emotional speech samples. In this work, we used the SAVEE (Surrey Audio-Visual Expressed Emotion) database and the Toronto Emotional Speech Set (TESS).

The SAVEE database was recorded from four native English speakers (identified as DC, JE, JK, KL), all postgraduate students and researchers at the University of Surrey, aged between 27 and 31. Emotions are psychologically described in seven discrete classes; for this study we employed five: anger, fear, happiness, sadness, and neutral speech. The textual material consisted of sentences selected from the standard TIMIT¹ corpus, phonetically balanced for each emotion.

The second emotional speech database is the Toronto Emotional Speech Set (TESS), a popular speech corpus widely used in research on emotion recognition from speech. This database was developed to enable the analysis of the effect of emotions on speech. It consists of recordings of twelve sentences spoken by professional speakers. The sentences were recorded to reflect seven basic emotions, of which we used five: happiness, sadness, anger, fear, and neutral. The speakers were selected to cover different age groups, enabling research into

¹ The TIMIT corpus of read speech was designed to provide speech data for acoustic-phonetic studies and for the development and evaluation of automatic speech recognition systems.

the effect of age on the perception and recognition of emotions. The database contains a total of 2,800 audio samples (12 sentences \times 7 emotions \times 2 speakers \times 100 repetitions), from which we used only those sentences corresponding to the aforementioned five emotional states. The recordings are provided in WAV format, with a standard sampling frequency of 44.1 kHz and 16-bit resolution. The sentences are short and standardized, which facilitates the analysis of emotional content. However, the recordings were made under ideal acoustic conditions, which may affect the model's performance in real-world noisy environments.

3. FEATURE EXTRACTION

Feature extraction from the speech signal plays a crucial role in the process of classifying emotional speech. The values of these features produce discriminative measures that determine the level of classification performance. Therefore, selecting the right set of features is essential to improve classification accuracy. In this study, we focus on the spectral properties of the speech signal.

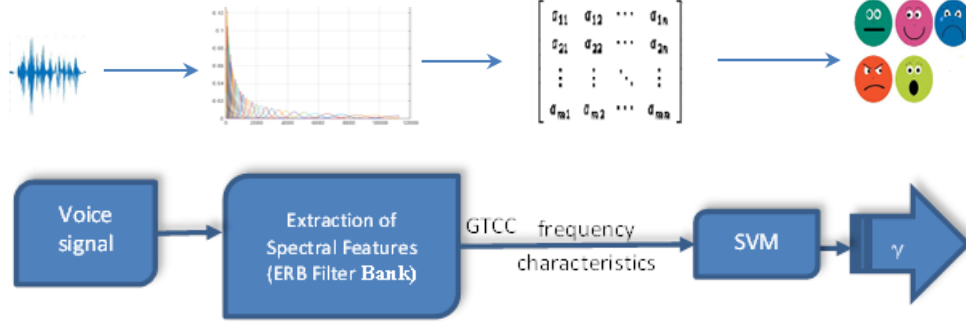


Fig. 1 Block diagram of the process of feature extraction and classification of emotional speech

Figure 1 illustrates the process of extracting spectral features from the speech signal using a gammatone filter bank based on the Equivalent Rectangular Bandwidth (ERB) scale, as well as certain frequency characteristics of speech. The spectral features are extracted from very short time frames of the speech signal (on the order of 20–30 ms), thus representing local characteristics. The GTCC spectral features capture only the short-term properties of the signal in the frequency domain. Finally, based on the resulting feature vector - composed of the aforementioned extracted spectral and frequency features the speech signal is classified into the corresponding emotional class γ .

3.1. Gammatone Filter bank

The gammatone filter bank is a set of filters used to analyze the frequency spectrum of the speech signal and is inspired by the way the human inner ear (cochlea) processes sound [2]. This approach models the mechanical and perceptual properties of the auditory system, which makes it useful for audio-processing tasks such as feature extraction for speech or speech-emotion recognition.

The basis of the gammatone filter bank is the gammatone function. The filters in the gammatone bank are based on the gammatone function, which is a time-domain function defined as follows [6]:

$$g(t) = t^{n-1} e^{-2\pi B t} \cos(2\pi f_c t + \phi). \quad (1)$$

where $g(t)$ is the impulse response of the gammatone filter, t is time, n is the filter order (typically 4 for modeling human hearing), B is the filter bandwidth (determined by the ERB - Equivalent Rectangular Bandwidth — scale), f_c is the filter center frequency, and ϕ is the signal phase. From the equation it follows that the function combines exponential decay with sinusoidal oscillation.

Gammatone filters are distributed uniformly along the ERB scale, which accurately models the resolution of human hearing. The ERB scale provides finer resolution at lower frequencies compared to linear or logarithmically spaced scales such as the Mel scale. The ERB bandwidth for a frequency f can be calculated using the following formula:

$$ERB = \left[\left(\frac{f_c}{EarQ} \right)^n + minBW^n \right]^{\frac{1}{n}}, \quad (2)$$

where $EarQ$ is the asymptotic quality factor of the filter at high frequencies, and $minBW$ is the minimum bandwidth at low frequencies. The filter quality (Q -factor) is the ratio of its center frequency to its bandwidth.

Several researchers have proposed different values for these parameters; however, the most widely accepted values were given in [7], where $EarQ = 1000 / (24.7 \cdot 4.37)$, $minBW = 24.7$, and $n = 1$. These parameter choices are mainly due to the larger quality factor achieved at lower frequencies. The ERB for a filter with center frequency f_c is defined as:

$$ERB(f) = 24.7 \cdot (4.37 \cdot f/1000 + 1). \quad (3)$$

The Equivalent Rectangular Bandwidth (ERB) is a psychoacoustic measure of the bandwidth of the auditory filter at each point along the cochlea [6] and corresponds to the bandwidth BBB in equation (1). The filter bank covers the desired frequency range of the signal (e.g., 20 Hz–8 kHz for speech). Each filter is centered on a specific center frequency and enables the analysis of the energy within that band.

Signal filtering: the input signal is convolved with each filter in the bank. The result is a set of time-domain signals that represent the energy in different frequency bands.

3.1.1. GTCC (Gammatone Cepstral Coefficients):

To compute GTCCs we first perform signal extraction. The speech signal is divided into segments of $N=512$ samples. The processing frames are 25 ms long and are overlapped in time with a hop (frame shift) of 10 ms (in our work). Next, each frame is windowed using a Hamming window. Windowing is the next step in the feature-extraction chain and serves to smooth and consolidate nearby spectral lines. We then compute the FFT (Fast Fourier Transform) for each window in order to obtain the spectrogram. The gammatone filter bank is applied to the spectrum (Figure 2): gammatone filters are used to filter the spectrum so that the filtered output represents the energy in each filter band. In our implementation the gammatone filter bank contains 40 filters. Frequency-domain analysis provides information about the energy distribution and formants, which are key cues for speech and emotion recognition.

To increase the dynamic range of the spectrum we take the logarithm of the energy - i.e., for each filter we compute the log-energy. Finally, applying the DCT to the vector of log-energies yields the GTCC coefficients equation (4). Typically one retains the first N coefficients; the choice of N depends on the intended application of the features (in our work we use 12 coefficients).

$$GTCC_m = \sqrt{\frac{2}{N}} \sum_{n=1}^N \log(X_n) \cos\left[\frac{\pi n}{N} \left(m - \frac{1}{2}\right)\right]. \quad (4)$$

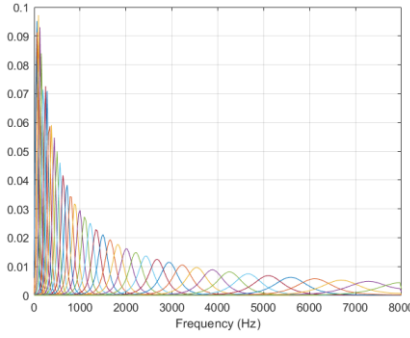


Fig. 2 ERB filter bank with amplitude response

In our feature vector we also include $\Delta GTCC$ and $\Delta^2 GTCC$, which help capture the signal dynamics and further improve emotion recognition. $\Delta GTCC$ are computed as follows:

$$\Delta C[t] = \frac{\sum_{n=1}^N n(C[t+n] - C[t-n])}{2 \sum_{n=1}^N n^2}. \quad (5)$$

where $C[t]$ is the GTCC at time step t , and N is the window size.

$\Delta^2 GTCC$ are given by the following equation:

$$\Delta^2 C[t] = \frac{\sum_{n=1}^N n(\Delta C[t+n] - \Delta C[t-n])}{2 \sum_{n=1}^N n^2}. \quad (6)$$

For the purpose of recognizing emotional states in the speech signal, we will use 12 GTCC, their first derivatives, and their accelerations (12 Δ and 12 Δ^2), which amounts to a total of 36 coefficients.

The reason we decided to use GTCC coefficients in the feature vector instead of MFCC is that GTCC coefficients are an improved, biologically inspired version of MFCC that better models human auditory perception, as well as their noise robustness and ability to preserve information in the low-frequency range, along with a better response for emotion classification.

Generally speaking, there are two main differences between MFCC and GTCC coefficients. The obvious difference lies in the frequency scale. GTCC, which is based on the Equivalent Rectangular Bandwidth (ERB) scale, has a finer resolution at low frequencies compared to MFCC (mel scale). The second difference is the nonlinear rectification step before applying the DCT (Discrete Cosine Transform). MFCC uses a logarithmic function, while GTCC uses a cube root.

Through careful examination of all the differences between MFCC and GTCC, it was concluded that the nonlinear rectification primarily accounts for the differences in noise robustness. Specifically, the cube root rectification provides greater feature robustness to noise compared to the logarithm. The reason for the cube root's better robustness may be that some speech data contains information across different energy levels. In a mixture with noise, there are dominant units or segments in the time-frequency (T - F) domain that carry information about this energy. The cube root operation makes the features scale-dependent (i.e., dependent on the energy level) and helps preserve this information. On the other hand, the logarithmic operation does not encode this information.

MFCC has been the most commonly used feature vector for emotional speech classification. However, MFCC systems usually do not perform well when noise is present in the speech signal. Noise distorts the extracted features, leading to an inconsistent probability calculation. Therefore, GTCC yields more accurate results than MFCC.

3.2 Frequency characteristics of speech

3.2.1. Spectral Centroid

The spectral centroid quantifies the "center of mass" of the spectrum of an audio signal and provides information about the dominant frequencies within a specific time window. It describes the "center of gravity" of the frequency spectrum of an audio signal and indicates where the dominant frequency in the signal is located. Intuitively, it is a measure that reflects whether a sound appears "bright" (higher frequencies) or "dark" (lower frequencies).

The spectral centroid is mathematically defined, as the center of gravity of the magnitudes by the following equation (7):

$$C = \frac{\sum_{i=1}^N f_i |X(f_i)|}{\sum_{i=1}^N |X(f_i)|} . \quad (7)$$

where f_i is the frequencies in the spectrum, $|X(f_i)|$ is the amplitude (magnitude) at the frequency f_i , N is the total number of frequencies. Different emotions, such as happiness, sadness or anger, can have different spectra in the speech signal, which affects the value of the spectral centroid. For example, anger contains a higher spectrum, which means a higher centroid, while sadness contains a lower spectrum, which means a lower centroid.

3.2.2. Spectral Decrease

Spectral Decrease is a measure that quantifies how much the intensity of frequency components decreases when moving from lower to higher frequencies. A low score indicates that there is dominant energy at higher frequencies, while a higher score indicates that the energy decreases toward higher frequencies. This frequency characteristic is based on spectral energy analysis. It can be used to identify the tonality or "sharpness" of speech. Equation (8) calculates the Spectral Decrease:

$$\text{SpectralDecrease} = \frac{\sum_{k=2}^N \frac{|X[k] - X[1]|}{k-1}}{\sum_{k=1}^N |X[k]|} . \quad (8)$$

where are they $|X[k]|$ frequency component amplitudes k , N total number of frequency components and $|X[1]|$ amplitude of the first (lowest) frequency component. This feature can be used to distinguish voices with pronounced fundamental frequencies (e.g. sad

speech) from voices with a "sharper" tone (e.g. anger) and helps identify emotional states that have specific spectral patterns.

3.2.3. Spectral Flux

Spectral Flux is a frequency characteristic that measures the change in spectral energy between successive time frames. This feature belongs to the group of dynamic frequency descriptors because it focuses on the variation in spectral energy over time, it provides an analysis of the change in spectral energy, focusing on the temporal change, making it useful for detecting dynamic aspects of speech. It is used to identify changes in pitch and intensity that are specific to emotional speech. The spectral flux is calculated according to the following formula (9):

$$\text{SpectralFlux} = \sum_{k=1}^N (X_{t+1}[k] - X_t[k])^2. \quad (9)$$

where $|X_t[k]|$ spectral amplitude of the frequency component k in the current time frame t , $|X_{t+1}[k]|$ spectral amplitude of the same frequency component in the next time frame $t+1$ i N total number of frequency components.

3.2.4. Spectral Spread

Spectral Spread belongs to frequency characteristics and is used to quantify the spectrum width of a signal. More precisely, it describes how far the frequency components of the spectrum are from its center, i.e. spectral centroid. Spectral Spread measures the dispersion of the frequency components of the spectrum around the spectral centroid. If the value of the spread is small, the energy of the spectrum is concentrated around the center, while the large value of the spread determines that the energy of the spectrum is spread over a wide frequency range. Equation (10) calculates the Spectral Spread:

$$\text{SpectralSpread} = \sqrt{\frac{\sum_{k=1}^N (f[k] - f_c)^2 |X[k]|^2}{\sum_{k=1}^N |X[k]|^2}}. \quad (10)$$

where $f[k]$ is frequency of the k -th component, f_c is the spectral centroid, $|X[k]|^2$ is the energy of the k -th frequency component and N represents the number of frequency components. In emotional speech, anger causes a larger spread due to high frequency components, while for example sadness results in a smaller spread due to the dominance of lower frequencies.

3.2.5. Spectral Rolloff Point

Spectral Rolloff Point belongs to the frequency characteristics and is used to quantify the energy content of the spectrum. It is a measure that identifies the frequency point below which a certain percentage (the typical value used is 85%, but can be different depending on the application) of the total energy of the spectrum. Let the spectral power of the signal be given by $P(f)$, where f is the frequency. The total spectrum energy is defined as in equation (11):

$$E_{total} = \sum_{f=0}^{f_{max}} P(f). \quad (11)$$

The spectral rolloff point $f_{rolloff}$ satisfies the condition of equation (12):

$$\sum_{f=0}^{f_{rolloff}} P(f) = \alpha E_{total} . \quad (12)$$

where f_{max} is maximum frequency, α is the selected threshold (e.g. 0.85 for 85% energy) and rolloff is the frequency point at which the threshold is met. Emotional speech affects the roll-off point, so that in anger the energy spreads towards higher frequencies, so the roll-off point is higher, while for example in sadness the energy is concentrated in lower frequencies, so the rolloff point is lower.

3.2.6. Spectral Slope

Spectral Slope belongs to the frequency characteristics and is used to describe the falling or rising nature of the signal spectrum. This feature quantifies the slope of the spectral curve, calculates the slope of the line that best approximates the amplitude spectrum in logarithmic frequency space, i.e. measures the relationship between frequency and spectrum amplitude. Spectral Slope is defined as in equation (13):

$$\text{Slope} = \frac{N \sum (f_k \cdot A_k) - \sum f_k \cdot \sum A_k}{N \sum (f_k^2) - (\sum f_k)^2} . \quad (13)$$

where f_k is the amplitude of the component k , A_k is the amplitude of the component k , N is the total number of components. A flat slope (close to zero) represents an even distribution of energy. A negative slope means that energy is concentrated in lower frequencies, while a positive slope means that energy increases towards higher frequencies. Anger or excitement may show a steeper negative slope due to more intense low frequencies and sadness may have a milder slope because the frequencies are more evenly distributed.

3.2.7. Spectral Crest

Spectral Crest belongs to the frequency characteristics and is used to describe the concentration of energy in the signal spectrum. This characteristic measures the ratio between the maximum amplitude of the spectrum and the total energy in the spectrum. Spectral Crest quantifies how much energy is concentrated in high frequencies in the spectrum. It relates to the distribution of energy across the frequency spectrum, as shown in equation (14):

$$\text{SpectralCrest} = \frac{\max(A)}{\sum(A)} . \quad (14)$$

where A is the amplitude of the frequency components. A high Spectral Crest indicates a greater concentration of energy in the higher frequencies. A low Spectral Crest indicates a more even distribution of energy across the spectrum, which is characteristic of melodic or tonal sounds like vowels. Anger and excitement often have high Spectral Crest values due to the increased concentration of energy in the higher frequencies. Sadness and calm emotions can have lower values due to a more even distribution of energy.

4. SUPPORT VECTOR MACHINE CLASSIFIER (SVM)

4.1. Definition SVM

Support Vector Machines (SVM) are one of the most powerful classification algorithms. The main idea of this algorithm is to find the optimal decision boundary (hyper-plane) that maximally separates the classes in the data [17]. The Support Vector Machine (SVM) algorithm performs classification by mapping a set of training samples from the input sample space R_N (input space) into an n -dimensional space F (feature space). It optimally separates the samples into two classes. Once the sample vector x is mapped into space F using the function Φ , the class to which the new vector $\Phi(x)$ belongs is determined in this new space. The hyper-plane separates the classes and represents a linear separating function. The hyper-plane is a geometric entity that divides the data space into two or more classes. The optimal hyper-plane is the one with the maximum separation margin between the samples of the two classes. It separates two separable subsets and is fully determined by specific vectors from both subsets, known as support vectors. The margin is the distance between the hyper-plane and the closest points from each class (known as support vectors). SVM aims to find the hyperplane with the maximum margin, thereby achieving better generalization of the model. Support vectors are a subset of data points that directly influence the position of the hyper-plane. All other points have no effect on defining the decision boundary. A hyper-plane in n -dimensional space can be represented as in equation (15):

$$wx + b = 0. \quad (15)$$

where w is the weight vector (hyper-plane direction), x input vector and b bias (hyper-plane offset).

The goal is to minimize the function given in equation (16):

$$\frac{1}{2} \|w\|^2, \quad (16)$$

with a restriction given in equation (17)

$$y_i(w \cdot x_i + b) \geq 1. \quad (17)$$

where $y_i \in \{-1, +1\}$ is the class label.

4.2. Soft margin classifier

In cases where the vectors are not linearly separable, meaning the training set contains some kind of "noise," a soft margin classifier is used. In this case, classification with a certain error in the training samples is allowed, with the goal of minimizing this error.

When the data is not perfectly separable, the parameter C is used to balance between maximizing the margin and minimizing errors given in equation (18):

$$\text{Minimiziraj: } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i, \quad (18)$$

so that equation (19) holds:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i \text{ za sve } i \geq 0, \quad (19)$$

where ξ_i is the size of the error for the i -th data.

4.3. Kernel functions

If it is not possible to separate the data using the support vector machine (SVM) with the soft margin approach, a so-called kernel function is employed. The basic idea of this method is the application of a certain function (Φ), which maps the basic or input vector space into a higher-dimensional space in which the data become linearly separable: $x \rightarrow \Phi(x)$.

All statements about SVM still hold in higher dimensional space. When constructing a linear hyperplane in a higher dimension space, when that plane is mapped back in the initial space, a nonlinear separation is obtained. The problem is to find the kernel function. Any function that satisfies Mercer's [15] theorem can represent a kernel function, that is, it can represent a scalar product in a vector space. Using kernels with non-linear classification problems are solved as linear ones (kernel trick) [14].

Several types of kernel functions are used by default:

- linear kernel given by equation (20):

$$K(x_i, x_j) = x_i^T x_j. \quad (20)$$

- polynomial kernel given by equation (21):

$$K(x_i, x_j) = (\gamma \cdot x_i^T x_j + r)^d, \gamma > 0. \quad (21)$$

- radial basis function (RBF) given by equation (22):

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0. \quad (22)$$

- sigmoidal kernel given by equation (23):

$$K(x_i, x_j) = \tanh(\gamma \cdot x_i^T x_j + r)^d. \quad (23)$$

where γ , r and d are kernel parameters [16] or hyperparameters. The given examples of kernel functions are sufficient in solving most classification problems.

5. PRACTICAL WORK AND RESULTS

In this section we describe how we performed the classification of speech utterances for the aforementioned emotional speech databases. The program for feature extraction and for classifying emotional utterances into the corresponding emotional groups was implemented in MATLAB R2023b. The first part of the code contains a module for loading emotional utterances from the databases. Next, we split the total set of utterances into two groups. The larger group, which is used for training the classifier, comprises 80% of the utterances, while the smaller group, used for testing the classifier, comprises the remaining 20%. Both groups contain an equal number of utterances for each emotional state. The next step in the classification pipeline is the feature extraction procedure from the speech signal, which is described in Section 3.

For feature extraction from the speech signal we used MATLAB's `audioFeatureExtractor` function. Processing frames were 25 ms long with 10 ms frame shift (i.e., frames were overlapped). The *ERB* scale was used for the extraction of the mentioned spectral features. The total number of feature vector dimensions is 44, consisting of 12 *gtcc*, 12 *gtccDelta*, 12 *gtccDeltaDelta* (36 GTCC-related coefficients) and 8 spectral features: *spectralCentroid*, *spectralDecrease*, *spectralFlux*, *spectralKurtosis*, *spectralSpread*, *spectralRolloffPoint*, *spectralSlope*, and *spectralCrest*.

After feature extraction, the next software module trains the classifier on the training set. During this step the classifier's model parameters are tuned in order to maximize classification performance. The remaining data, referred to as the test set, are used to estimate the error and to measure the classifier's success on unseen data. Note that the test error depends on the random partitioning of the corpus into training and test sets, and there is also a risk of overfitting (excessive model adaptation to the training data).

To avoid such an undesirable situation, we used the *k-fold cross-validation* method. This method randomly splits the available dataset into k equally sized parts, i.e. of length n/k , where n is the total number of available samples. The chosen classifier is then trained k times, each time using a different part as the test set. The overall error is the average of the errors over the iterations. By randomly partitioning into k equal parts while preserving class proportions, we obtain a *stratified k-fold cross-validation*, which ensures an adequate representation of each class in the folds. In our work we used *10-fold cross-validation*.

Our classifier model is an SVM based on the RBF (*Gaussian*) kernel (equation (23)), where γ denotes the kernel scale parameter (*KernelScale*). The *BoxConstraint* corresponds to the regularization parameter C and it was set to 1. This parameter controls the „hardness“ of the margin: larger values of C can lead to overfitting, while smaller values increase tolerance to misclassification.

The parameter that controls the width of the Gaussian kernel—commonly denoted γ , which is critical for the performance of a Gaussian SVM. The Standardize option was set to true so that MATLAB automatically standardizes the feature values prior to model training. To achieve the best possible performance, hyperparameter tuning is recommended. In MATLAB we used *fitcecoc* with automatic hyperparameter optimization to find the optimal pair (*BoxConstraint*, *KernelScale*).

For classification of emotional speech utterances we evaluated both an SVM without hyperparameter optimization and an SVM with hyperparameter optimization. The hyperparameter-optimized classifier demonstrated substantially improved classification performance, as evidenced in the following figures and tables presented in the paper.

Here are the classification results for the Toronto Emotional Speech Set (TESS) using an SVM classifier without hyper-parameter optimization.

Figure 3 shows the confusion matrix for the classification of speech utterances from the TESS database into their corresponding emotional classes, without hyper-parameter optimization. The validation accuracy of the classifier was 76.375%, while the model's accuracy was 80.5%.

True Class	Anger	67	6	4	2	1	83.8%	16.2%
	Fear	6	64	5	5		80.0%	20.0%
	Happiness	7	5	60	7	1	75.0%	25.0%
	Neutral	8	7	4	61		76.2%	23.8%
	Sadness	2	4	3	1	70	87.5%	12.5%
		74.4%	74.4%	76.9%	80.3%	97.2%		
		25.6%	25.6%	21.1%	19.7%	2.8%		
		Anger	Fear	Happiness	Neutral	Sadness		
		Predicted Class						

Fig. 3 Classification accuracy of the SVM model on the TESS dataset without hyper-parameter optimization (Confusion Matrix)

True Class	Anger	78		1	1		97.5%	2.5%
	Fear	1	76	3			95.0%	5.0%
	Happiness	4	1	73	1	1	91.2%	8.7%
	Neutral	2	1	1	76		95.0%	5.0%
	Sadness		2			78	97.5%	2.5%
		91.8%	95.0%	93.6%	97.4%	98.7%		
		8.2%	5.0%	6.4%	2.6%	1.3%		
		Anger	Fear	Happiness	Neutral	Sadness		
		Predicted Class						

Fig. 4 Classification accuracy of the SVM model on the TESS dataset with hyper-parameter optimization (Confusion Matrix)

Figure 4 shows the confusion matrix for the classification of utterances from the TESS speech corpus obtained with hyper-parameter optimization. The classifier achieved a validation accuracy of 91.6875%, while the final model accuracy was 95.25%.

The MATLAB function `fitcecoc` was used to train SVM classifiers for multi-class classification via the Error-Correcting Output Codes (*ECOC*) strategy. This function supports hyperparameter optimization, including tuning of *BoxConstraint* and *KernelScale*.

The *BoxConstraint* hyperparameter (the regularization parameter C) controls the trade-off between training accuracy and generalization to unseen data: a high *BoxConstraint* allows a more complex decision boundary that emphasizes correct classification on the training set but may lead to overfitting, whereas a low *BoxConstraint* yields a simpler boundary that tolerates some training errors and is generally more robust to noise and better at generalizing.

The *KernelScale* hyper-parameter scales the RBF (*radial basis function*) kernel: larger values correspond to a wider kernel (smoother, more global decision regions), while smaller values focus the model on more local structure in the data. The maximum number of hyperparameter optimization iterations was set to 40.

The classification of speech utterances for the SAVEE (Surrey Audio-Visual Expressed Emotion) database without hyper-parameter optimization is given by the confusion matrix in Figure 5.

True Class	Anger	11		1			91.7%	8.3%
	Fear		11			1	91.7%	8.3%
	Happiness			10		2	83.3%	16.7%
	Neutral			1	11		91.7%	8.3%
	Sadness					12	100.0%	
		100.0%	100.0%	83.3%	100.0%	80.0%		
				16.7%		20.0%		
		Anger	Fear	Happiness	Neutral	Sadness		
		Predicted Class						

Fig. 5 Classification accuracy of the SVM model on the *SAVEE* dataset without hyper-parameter optimization (Confusion Matrix)

Following the classification process, the validation accuracy of the classifier was 93.75%, and the model accuracy was 94.5833%.

True Class	Anger	12					100.0%	
	Fear		11			1	91.7%	8.3%
	Happiness			11		1	91.7%	8.3%
	Neutral		1		11		91.7%	8.3%
	Sadness					12	100.0%	
		100.0%	91.7%	100.0%	100.0%	85.7%		
			8.3%			14.3%		
		Anger	Fear	Happiness	Neutral	Sadness		
		Predicted Class						

Fig. 6 Classification accuracy of the SVM model on the *SAVEE* dataset with hyper-parameter optimization (Confusion Matrix)

Figure 6 shows the confusion matrix for the classification of speech utterances from the *SAVEE* database with hyper-parameter optimization, resulting in a classifier validation accuracy of 96.6667% and a model accuracy of 95.0000%.

After classification, we evaluated the classifier's performance using the following measures: Accuracy (Accuracy measures the overall percentage of correctly classified samples, i.e., how often the model was correct), equation (25); Precision (Precision measures, of all samples classified as positive, how many truly belong to the positive class), equation (26); Recall (Recall measures how many of the truly positive samples the model successfully detected), equation (27); and the F1 score (the F1 score is the harmonic mean of precision and recall), equation (28).

$$\text{Accuracy} = \frac{\text{Number of correctly classified samples}}{\text{Total number of samples}} = \frac{TP+TN}{TP+TN+FP+FN} \quad (25)$$

where TP (True Positives) correctly classified positive samples; TN (True Negatives) correctly classified negative samples; FP (False Positives) negative samples incorrectly classified as positive; and FN (False Negatives) positive samples incorrectly classified as negative.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (26)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (27)$$

$$\text{F1 Score} = 2 \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (28)$$

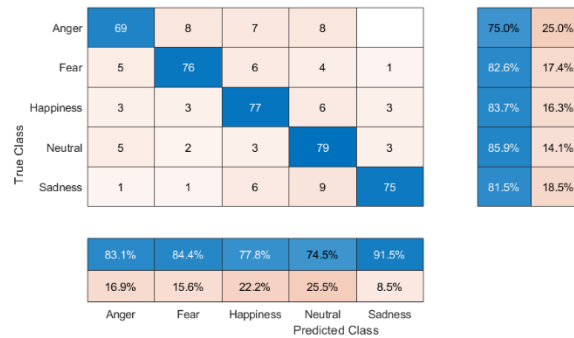
Table 1 Performance evaluation of the classifier for the TESS database with hyper-parameter optimization

Emotional class	Precision by class	Recall by class	F1-score by class	Accuracy in %
'Anger'	0.9176	0.9750	0.9455	0.9775
'Happiness'	0.9500	0.9500	0.9500	0.9800
'Sadness'	0.9359	0.9125	0.9241	0.9700
'Fear'	0.9744	0.9500	0.9620	0.9850
'Neutral'	0.9873	0.9750	0.9600	0.9925
Total:	0.95305	0.95250	0.95253	0.98100

Table 2 Performance evaluation of the classifier for the SAVEE database with hyper-parameter optimization

Emotional class	Precision by class	Recall by class	F1-score by class	Accuracy in %
'Anger'	1.0000	1.0000	1.0000	1.0000
'Happiness'	0.9167	0.9167	0.9167	0.9667
'Sadness'	1.0000	0.9167	0.9565	0.9833
'Fear'	1.0000	0.9167	0.9565	0.9833
'Neutral'	0.8571	1.0000	0.9231	0.9667
Total:	0.95476	0.95000	0.95056	0.98000

We performed classification without classifier hyper-parameter optimization on the speech utterances from both databases (SAVEE and TESS) and obtained a confusion matrix (Figure 7). The validation accuracy of the classifier was 75.7609%, while the model accuracy was 81.7391%.

**Fig. 7** Classification accuracy of the SVM model on the *SAVEE*+ TESS dataset without hyper-parameter optimization (Confusion Matrix)

After optimizing the hyper-parameters for the SVM classifier (*BoxConstraint* and *KernelScale*), we obtained a confusion matrix (Figure 8) and a validation accuracy of 93.0435%, with a model accuracy of 95.2174%.

True Class	Anger	87	5				94.6%	5.4%
	Fear	3	86	2	1		93.5%	6.5%
	Happiness	1	2	88		1	95.7%	4.3%
	Neutral	3	1		88		95.7%	4.3%
	Sadness		1	1	1	89	96.7%	3.3%
		92.6%	90.5%	96.7%	97.8%	98.9%		
		7.4%	9.5%	3.3%	2.2%	1.1%		
		Anger	Fear	Happiness	Neutral	Sadness		
		Predicted Class						

Fig. 8 Classification accuracy of the SVM model on the *SAVEE*+ TESS dataset with hyper-parameter optimization (Confusion Matrix)

Table 3 Performance evaluation of the classifier for the *SAVEE*+TESS database with hyper-parameter optimization

Emotional class	Precision by class	Recall by class	F1-score by class	Accuracy in %
'Anger'	0.9255	0.9457	0.9355	0.9739
'Happiness'	0.9053	0.9348	0.9198	0.9674
'Sadness'	0.9670	0.9565	0.9617	0.9848
'Fear'	0.9778	0.9565	0.9670	0.9870
'Neutral'	0.9889	0.9674	0.9780	0.9913
TOTAL:	0.9529	0.95217	0.95241	0.98087

Table 3 presents the performance evaluation results of the classifier for the combined *SAVEE*+TESS speech utterance databases.

Few studies have addressed the construction of feature vectors for training and testing emotion recognition and classification from speech signals using GTCC in combination with other features. Our results clearly demonstrate that the proposed classifier achieved superior emotion recognition performance on speech utterances compared to the works reported in [22] and [23]. Furthermore, it exhibited higher classification accuracy than approaches based on advanced deep learning architectures, such as the Deep Convolutional Recurrent Neural Network (Deep C-RNN) applied to composite feature sets comprising Mel-Frequency Cepstral Coefficients (MFCC) and Gammatone Frequency Cepstral Coefficients (GFCC) [24].

6. CONCLUSION

This paper presents an SVM-based emotional speech classification model whose hyper-parameters are optimized during training. The feature vector comprises spectral and frequency-based characteristics of the speech signal. The classification model uses GTCC together with their first and second derivatives (Δ GTCC and Δ^2 GTCC), which represent short-term spectral properties of the signal, alongside selected frequency-domain speech features that we believe strongly contribute to improved emotion classification. These coefficients condense frequency-domain information into a compact, discriminative representation, which is crucial for emotion recognition. Frequency-domain analysis

provides information about the energy distribution, which we exploit to recognize emotions. Fundamentally, we are interested in the distribution of energy in the speech signal because our classification is based on how something is said rather than what is said. The proposed classifier was validated on two separate datasets (SAVEE and TESS) and on a combined dataset (SAVEE+TESS), and it demonstrated solid classification performance for emotional utterances.

Future work will focus on developing a hybrid emotion-classification model that integrates acoustic and frequency-domain features. The model will consist of two parallel branches for feature extraction and classification. The first branch will use a feature vector composed of GTCC coefficients and the first three formants (F1–F3), and classification will be performed by an SVM with optimized hyper-parameters. The second branch will employ a prosodic feature vector (e.g., fundamental frequency, intensity, duration, ...) as input to a neural network designed for classification. The final decision will be obtained by decision-level fusion of the two classifiers' outputs. We expect that the proposed hybrid architecture, through the complementarity of the features, will yield measurable improvements in accuracy compared to unimodal approaches.

A second research direction involves the development of a multimodal model for emotional speech classification. This approach would integrate acoustic speech features (e.g., spectral and prosodic) with facial-expression features. The synergistic combination of these complementary modalities would enable the model to overcome the limitations of unimodal systems and is expected to result in significant improvements in accuracy and robustness.

To improve the robustness of emotion classifiers in real-world noisy environments, the training database needs to be substantially expanded. Key directions for improvement include: (1) diversifying the speech corpus with a larger number of samples, words, and sentences; (2) explicitly including recordings with background noise; and (3) recruiting a larger and more diverse set of speakers so the model can better generalize across different voices.

REFERENCES

- [1] Minu Babu et al "Whether MFCC or GFCC is better for recognizing emotion from speech?", International journal of research in computer applications and robotics Vol.2 Issue.6, Pg.: 14-17, June 2014, www.ijrcar.com
- [2] Holdsworth J, Smith I,N, Patterson R, Rice P. Implementing a Gammatone filter bank. Annex C of the SVOS final report: Part A: The auditory filterbank. 1988. p. 1–5. Available from: <https://www.pdn.cam.ac.uk/other-pages/cnbh/files/publications/SVOSAnnexC1988.pdf>
- [3] O. Cheng, W. Abdulla, Z. Salcic, and N. Zealand, "Performance Evaluation of Front-End Algorithms for Robust Speech Recognition," Signal Processing and Its Applications, 2005. Proceedings of the Eighth International Symposium on, vol. 2, pp. 711–714, 2005.
- [4] [52] R. Schl, I. Bezrukov, H. Wagner, and H. Ney, "Gammatone Features and Feature Combination for Large Vocabulary Speech Recognition," Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on, vol. 4, pp. 649–652, 2007.
- [5] [54] X. Valero, S. Member, and F. Alias, "Gammatone Cepstral Coefficients: Biologically Inspired Features for Non-Speech Audio Classification," Multimedia, IEEE Transactions on, vol. 14, no. 6, pp. 1684–1689, 2012.
- [6] M. Slaney, "An Efficient Implementation of the Auditory Filter Bank," Apple Computer, Perception Group, Tech. Rep, 1993.
- [7] B. R. Glasberg and B. C. Moore, "Derivation of auditory filter shapes from notched-noise data," Hearing research, vol. 47, no. 1, pp. 103–138, Aug. 1990.
- [8] Utane, Akshay S., and S. L. Nalbalwar. "Emotion Recognition through Speech Using Gaussian Mixture Model and Support Vector Machine."emotion 2 (2013): 8.

- [9] P. Boersma, "Accurate Short-Term Analysis of the Fundamental Frequency and the Harmonics-to-Noise Ratio of a Sampled Sound," IFA Proceedings, (17), 1993.
- [10] Martijn Goudbeek, Jean Philippe Goldman, Klaus R. Scherer, „Emotion dimensions and formant position“, https://bridging.uvt.nl/pdf/goudbeek_goldman_scherer_interspeech_2009.pdf
- [11] Ververidis, Dimitrios, and Constantine Kotropoulos. "Emotional speech recognition: Resources, features, and methods." *Speech communication* 48.9 (2006): 1162-1181.
- [12] 11. Huang Y, Ao W, Zhang G (2017) Novel sub-band spectral centroid weighted wavelet packet features with importance-weighted support vector machines for robust speech emotion recognition. *Wireless Pers Commun* 95(3):2223–2238.
- [13] Mao Q, Xu G, Xue W, Gou J, Zhan Y (2017) Learning emotiondiscriminative and domain-invariant features for domain adaptation in speech emotion recognition. *Speech Commun* 93:1–10
- [14] Jordan, M.: The Kernel Trick, *Advanced Topics in Learning & Decision Making*, Berkeley, 2004.
- [15] Minh, H.; Q.; Niyogi, P.; Yao, Y.: Mercer Theorem, Feature Maps, and Smoothing, *Lecture Notes in Computer Science*, Springer Berlin, 2006
- [16] Cortes, C.; Vapnik, V.: Support Vector Networks, *Machine Learning*, vol.20, pp. 273-297, Kluwer Academic Publishers, Boston, 1995.
- [17] Pan, Yixiong, Peipei Shen, and Liping Shen. "Speech emotion recognition using support vector machine." *International Journal of Smart Home* 6.2 (2012): 101-108.
- [18] D. Ververidis, C. Kotropoulos, I. Pitas, Automatic emotional speech classification, In. *Proc. 2004 IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 1, pp. 593-596, Montreal, 2004.
- [19] Pan, Yixiong, Peipei Shen, and Liping Shen. "Speech emotion recognition using support vector machine." *International Journal of Smart Home* 6.2 (2012): 101-108.
- [20] Prabhakar GA, Basel B, Dutta A, Rao CVR (2023) Multichannel cnn-blstm architecture for speech emotion recognition system by fusion of magnitude and phase spectral features using DCCA for consumer applications. *IEEE Transactions on consumer electronics*
- [21] Hama Saeed M (2023) Improved speech emotion classification using deep neural network. *Circuits Syst Signal Proc* 42(12):7357–7376
- [22] Alluhaidan AS, Saidani O, Jahangir R, Nauman MA, Neffati OS (2023) Speech emotion recognition through hybrid features and convolutional neural network. *Appl Sci* 13(8):4750
- [23] Manuel Cardona, Vijender K. Solanki, Speech emotion recognition using gammatone cepstral coefficients and deep learning features, *Proceedings of the 2023 IEEE International Conference on Machine Learning and Applied Network Technologies*
- [24] U. Kumaran, S. Radha Rammohan, Senthil Murugan Nagarajan, A. Prathik, Fusion of mel and gammatone frequency cepstral coefficients for speech emotion recognition using deep C-RNN, June 2021 *International Journal of Speech Technology* 24(2), Volume 24, pages 303–314, (2021)