IUST
Iran University of
Science and Technology

Iranian Journal of Electrical and Electronic Engineering

Journal Homepage: ijeee.iust.ac.ir

A Novel Nonparametric Kernel for Speech Emotion Recognition

M. Hasheminejad *(C.A.)

Abstract: The Nonparametric Speech Kernel (NSK), a nonparametric kernel technique, is presented in this study as a novel way to improve Speech Emotion Recognition (SER). The method aims to effectively reduce the size of speech features to improve recognition accuracy. The proposed approach addresses the need for efficient and compact low-dimensional features for speech emotion recognition. Having acknowledged the intrinsic distinctions between speech and picture data, we have refined the Kernel Nonparametric Weighted Feature Extraction (KNWFE) formulation to suggest NSK, which is especially intended for speech emotion identification. The output of NSK can be used as input features for deep learning models such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), or hybrid architectures. In deep learning, NSK can also be used as a kernel function for kernel-based methods such as kernelized support vector machines (SVM) or kernelized neural networks. Our tests demonstrate that NSK outperforms current techniques, outperforming the best-tested approach by 5.02% and 3.05%, respectively, with an average accuracy of 96.568% for the Persian speech emotion dataset and 82.56% for the Berlin speech emotion dataset.

Keywords: Nonlinear Kernel, Speech, Emotion Recognition, Machine Learning, Feature Extraction, Support Vector Machine.

1 Introduction

EMOTION has been studied in several scientific fields such as biology (physiology), psychology, speech science, neuroscience, psychiatry, anthropology, sociology, communication, etc. Issues such as business management and advertising require extensive use of emotion processing. Consequently, given the complexity and variety of emotions, distinct views are expressed on this concept. Emotion psychology can be understood as a multifaceted interaction between conduct (speaking combined with body language) and consciousness (psychology). Emotion, in general, represents a mental state that arises spontaneously, an expressive situation, and neuronal activity. Researchers showed that there are more than 300 different emotions. Most of them introduced emotions such as anger, disgust, fear, joy, sadness, and wonder as basic ones. Emotions in human beings contain physiological stimulation, meaningful behavior, and conscious experiences. Therefore, there are three aspects of natural emotions, including psychological (what you think), physiological (what the body does), and expressive (how it reacts) [1].

Speech Emotion Recognition (SER) is a branch of Automatic Speech Recognition (ASR), which uses signal, feature extraction processes, and machine learning techniques like deep learning (DL) architectures. SER research has been integrated with existing works in ASR on feature extraction, such as the use of Mel Frequency Cepstral Coefficients (MFCC) for classification and pattern recognition. DL approaches developed in ASR or NLP, such as convolutional neural networks (CNN) and recurrent neural networks (RNN), are evaluated for SER practical applications. SER is expected to be a key technology in the development of innovative Human-Computer Interaction (HCI), Human-Machine Interaction (HMI), Human-Robot Interaction (HRI) systems, and affective computing, particularly in the upcoming era of the Internet of Things (IoT) [2].

The application of emotional information in a

Iranian Journal of Electrical & Electronic Engineering, 2023.
Paper first received 30 May 2022 and accepted 13 Dec 2023.

* The author is with the Department of Electrical Engineering, University of Jiroft, Jiroft, Iran.
E-mail: mhn@ujiroft.ac.ir.
Corresponding Author: M. Hasheminejad.

speech conversation system can increase user-friendliness. It also provides a more natural way for human-computer interaction that is achievable using an emotion recognition system in which the features are usually introduced by a spectral range of dominant information while ignoring the use of phase. The Fisher kernel exploits phase-based features. The Fisher kernel (equation (1)) uses stochastic models in decision-boundary classifiers such as SVM. It can transform a set of low-level descriptors into a fixed-length vector in which its dimensions depend on the size of the model parameters rather than the length of the sample vector (input data like X).

$$K\left(X_i, X_j\right) = \Phi(X_i)^T I^{-1} \Phi(X_j) \qquad (1)$$

Nonparametric Weighted Feature Extraction is a method for high dimensional multiclass pattern recognition that is a nonparametric extension of the scatter matrix. Using the proposed nonparametric scatter matrices, has at least two advantages. First, since it is generally of full rank, it can specify the desired number of extracted features and reduce the effect of the singularity problem. Second, the nonparametric nature of scattering matrices can apply the method to non-Gaussian data. This method assigns more weights to the samples near the decision boundary. This assignment leads to preserving the structures of decision-making boundaries better and finally improves the classification accuracy [3]. In particular, we redesigned the formulation of this non-parametric kernel to recognize the emotion from the speech signal and optimize its performance.

The rest of the paper is organized as follows: Section 2 presents an overview of the related works. The proposed kernel method is explained in section 3, and experimental results are presented in section 4. We conclude the paper in section 5 and explain some future directions.

2 Related Works

Figure 1 shows the proposed nine-level S^{2}C^{2}RASCMLI topology. To attain the desired voltage level, the H-bridge is inserted in the backyard to change the polarity and then two capacitors are connected in series.

In many previous studies, global acoustic features were for dimensionality reduction [4]. However, the emotional information of speech is determined by its dynamic changes. In other words, the emotional components often change with time instead of being fixed on a sentence. As a result, when global features are used alone in a phrase, local dynamic information of emotion in a speech signal may be overlooked [5]. By segmenting speech, one can easily avoid a lack of global features while considering such local features [6]. In [7], several methods have been proposed for segmentation. According to the results, an ensemble of global features and time-interval-dependent features significantly improved the results.

[8] proposes combining neural-responses-based and conventional acoustic-property-based approaches to improve emotion recognition from noisy speech. The proposed feature is obtained from an Auditory-nerve simulation model, while traditional acoustic-property-based features are provided by INTERSPEECH 2010 speech dataset. The motivation of the research is that the nervous responses are resistant to noise.

Many features are usually extracted from the speech signal to prevent the loss of information. The large size makes these features difficult to use. Therefore, feature reduction methods will be helpful here. To this end, kernel-based methods have been widely used [9]. Multi-kernel dimensionality reduction has been studied to obtain valuable information for emotion recognition to improve the performance of identifying emotion from speech. Therefore, a multi-kernel learning plan is usually an additional mapping of several kernels, where implementing this method may result in losing valuable information. A two-dimensional design that learns a two-dimensional subspace is proposed to solve the problem. This scheme provides a large number of linear combinations based on multi-kernel learning without a non-negative limitation so that more information is retained in this learning method. Multi-kernel learning and two-dimension subspace are then combined and made a unit construction. Generalized Multiple Kernel Discrimination Analysis is proposed to recognize emotion from speech. In this scheme, it is tried to have the possibility to access several mapping paths for multi-kernel combinations, and it does not include an irreversible limitation to achieve an unrealistic map [10].

3 Proposed method

Inspired by NWFE [3], The proposed framework utilizes an enhanced Nonparametric Speech Kernel (NSK) for emotion recognition. The following explains the proposed NSK method.

3.1 Nonparametric weighted feature extraction

The steps of the NWFE algorithm are as follows: The distance between each sample pair is first calculated and entered into a matrix known as the distance matrix; this distance is an Euclidean distance. The distance matrix is then used to calculate weight matrices (equation (2)). The average weighted matrix (equation (3)) and also scatter matrix weight (equation (4)) are calculated by using the weight matrix. Finally, the within-class (equation (5)) and between-class (equation (6)) scatter matrices are calculated. The conversion matrix is then built using the eigenvalue and eigenvector decompositions [3].

$$W_{lk}^{(i,j)} = \frac{dist(x_l^{(i)}, x_k^{(j)})^{-1}}{\sum_{t=1}^{n_j} dist(x_t^{(i)}, x_k^{(j)})^{-1}} \tag{2}$$

$$M_j\left(x_l^{(i)}\right) = \sum_{k=1}^{N_j} W_{lk}^{(i,j)} x_k^{(j)} \tag{3}$$

$$\lambda_l^{(i,j)} = \frac{dist(x_l^{(i)}, M_j(x_l^{(i)}))^{-1}}{\sum_{t=1}^{N_j} dist(x_t^{(i)}, M_j(x_t^{(i)}))^{-1}} \tag{4}$$

$$S_w = \sum_{i=1}^{L} P_i \sum_{l=1}^{N_i} \frac{\lambda_l^{(i,j)}}{n_i} (x_l^{(i)} - M_j(x_l^{(i)})(x_l^{(i)} - M_j(x_l^{(i)}))^t \tag{5}$$

$$S_b = \sum_{i=1}^{L} P_i \sum_{\substack{j=1 \\ j\neq i}}^{L} \sum_{l=1}^{N_i} \frac{\lambda_l^{(i,j)}}{n_i} (x_l^{(i)} - M_j(x_l^{(i)})(x_l^{(i)} - M_j(x_l^{(i)}))^t \tag{6}$$

In these equations, x_l^i, is the l^{th} sample of class i, N_i is the number of training samples of class i, and p_i is the prior probability of class i.

3.2 Kernel nonparametric weighted feature extraction

KNWFE is a nonlinear kernel of the NWFE. The strategy of kernel methods is to map data from original space to a Hilbert space of higher dimension, where the data is expected to have more feature separability. In this method, the internal multiplication of samples in the feature space can be calculated directly from the original data using the kernel function [11].

The main idea of using kernel is the nonlinear mapping of input data from original space into an appropriate feature space, where a kernel function can compute the inner product in the feature space. For example, equation (7) shows a linear kernel, equation (8) depicts a polynomial kernel, and the Gaussian kernel is shown in equation (9) [11].

$$\kappa(x,z) = \langle x,z \rangle \tag{7}$$

$$\kappa(x,z) = (\langle x,z \rangle + 1)^r \ , \ r \in Z^+ \tag{8}$$

$$\kappa(x,z) = \exp\left(\frac{-\|x-z\|^2}{2\sigma^2}\right) \tag{9}$$

where x and z are samples in R^d.

The size of the kernel matrix is an N×N, where N is the total number of samples. The weight matrix and the scatter-matrix weight are then calculated according to the kernel matrix. Then, within-class and between-class scatter matrices are calculated using those matrices. Henceforth, the converted matrix is built using the decomposition of the eigenvalue and eigenvector and solving a set of related equations. In this stage, the matrix reduces the data dimension. If A is a convert matrix, the final equation of feature reduction with this algorithm will be as equation (10) [11]:

$$Y = A^T \begin{bmatrix} \kappa(x_1, z) \\ \vdots \\ \kappa(x_N, z) \end{bmatrix} \tag{10}$$

The boundary data is more informative in classification since they determine the boundary of each class. These two algorithms try to increase the impact of the boundary data by assigning more weight to them.

Classification is the last step of identifying emotion from the speech signal. By this, different emotions can be recognized from speech. This step has two main phases: training and testing the classifier. The classifier is formed using the training algorithm and data in the training stage, and its performance is evaluated in the test stage. Through this, the generalization ability of the classifier is tested, and we can conclude whether the classifier has suitable performance towards unseen data (data not in the training set).

The KNWFE algorithm works on the principle of data weighting. Each sample is given a weight, which is assigned based on data of the same class as well as other classes. The KNWFE algorithm works on the principle of data weighting. Each sample is given a weight, which is assigned based on data of the same class as well as other classes. These weights are used to determine the scatter matrix weight. The scatter matrix is representative of the membership degree of each data to its class and other classes. The elements of this matrix is indicated by $\lambda_l^{(i,j)}$. In the proposed strategy, we take two significant points into account.

The first is that if the datum is near the center of the class, it can get more weight. As moving away from the center, the corresponding weight also decreases. In the proposed method assigns more weight to the boundary data and less to the central ones. An important point here is that under these conditions, noise-infected data that are far from the center will gain more weight, and this may lead to boundary misplacement. We use a Riley distribution function to fix the problem, which means the proposed algorithm passes the scatter matrix elements through this function. The second point is that the weights of near and far data should not be equal. The proposed approach modifies the degree of belonging of data in the matrix $\lambda_l^{(i,j)}$.

In the KNWFE algorithm, which is a nonlinear method based on the kernel of NWFE, the data weight matrix is defined as follows [11].

$$\lambda_l^{(i,j)} = \frac{\left[K_{ll}^{(i,i)}+\left(W^{(i,j)}K^{(j,j)}{W^{(i,j)}}^T\right)_{ll}-2\left(K^{(i,j)}{W^{(i,j)}}^T\right)_{ll}\right]^{-\frac{1}{2}}}{\sum_{t=1}^{N_i}\left[K_{tt}^{(i,i)}+\left(W^{(i,j)}K^{(j,j)}{W^{(i,j)}}^T\right)_{tt}-2\left(K^{(i,j)}{W^{(i,j)}}^T\right)_{tt}\right]^{-\frac{1}{2}}} \quad (11)$$

where l represents data index, i=1,2,…,L and j=1, 2, ..., L, that L is the number of classes, $K^{(i,j)}$ is a part of kernel matrix K, and $W^{(i,j)}$ is the weight matrix which is defined in equation (11), and matrix $\Lambda^{(i,j)}$ is defined in equation (12) [11]:

$$W^{(i,j)} = \begin{bmatrix} w_{11}^{(i,j)} & \cdots & w_{1N_j}^{(i,j)} \\ \vdots & \ddots & \vdots \\ w_{N_i1}^{(i,j)} & \cdots & w_{N_iN_j}^{(i,j)} \end{bmatrix} \quad (11)$$

$$\Lambda^{(i,j)} = diag\left\{\frac{\lambda_1^{(i,j)}}{N_i}, \dots, \frac{\lambda_{N_i}^{(i,j)}}{N_i}\right\} \quad (12)$$

$w_{lk}^{(i,j)}$ is defined in equation (13). N is the total number of data, and N_i is the number of data in the i^{th} class.

$$w_{lk}^{(i,j)} = \frac{(K_{ll}^{(i,i)} + K_{kk}^{(j,j)} - 2K_{lk}^{(i,j)})^{-1}}{\sum_{t=1}^{N_j}(K_{ll}^{(i,i)} + K_{tt}^{(j,j)} - 2K_{lt}^{(i,j)})^{-1}} \quad (13)$$

Matrices W and B are calculated as follows (equation (14) to (21)) to obtain the conversion matrix.

$$W = W_1 - W_2 - W_2^T + W_3 \quad (14)$$

$$W_1 = diag\{P_1\Lambda^{(1,1)}, \dots, P_L\Lambda^{(L,L)}\} \quad (15)$$

$$W_2 = diag\{P_1\Lambda^{(1,1)}W^{(1,1)}, \dots, P_L\Lambda^{(L,L)}W^{(L,L)}\} \quad (16)$$

$$W_3 = diag\{P_1{W^{(1,1)}}^T\Lambda^{(1,1)}W^{(1,1)}, \dots, P_L{W^{(L,L)}}^T\Lambda^{(L,L)}W^{(L,L)}\} \quad (17)$$

$$B = B_1 - B_2 - B_2^T + B_3 \quad (18)$$

$$B_1 = diag\left\{P_1\sum_{j=1}^{L}\Lambda^{(1,j)}, \dots, P_L\sum_{j=1}^{L}\Lambda^{(L,j)}\right\} \quad (19)$$

$$B_2 = \begin{bmatrix} P_1\Lambda^{(1,1)}W^{(1,1)} & \cdots & P_1\Lambda^{(1,L)}W^{(1,L)} \\ \vdots & \ddots & \vdots \\ P_L\Lambda^{(L,1)}W^{(L,1)} & \cdots & P_L\Lambda^{(L,L)}W^{(L,L)} \end{bmatrix} \quad (20)$$

$$B_3 = \sum_{i=1}^{L} p_i diag\{{W^{(i,1)}}^T\Lambda^{(i,1)}W^{(i,1)}, \dots, {W^{(i,L)}}^T\Lambda^{(i,L)}W^{(i,L)}\} \quad (21)$$

p in the above equation is the possibility of each class. B. C. Kuo et al. suggested the following steps in [11] to get the conversion matrix A:

If the conversion matrix is achieved based on the Fisher equation as equation (22), it is necessary to analyze the eigenvalue and eigenvector to get P and U.

$$A = PU \quad (22)$$

P is the eigenvector of the kernel matrix, and U is the eigenvector of the equation (24). These eigenvectors are ordered based on the greatest eigenvalue. Following that, the algorithm eliminates the eigenvectors with the corresponding eigenvalue of zero.

$$K = P\Gamma P^T \quad (23)$$

$$(\Gamma P^T(B - W)P\Gamma)U = \lambda(\Gamma P^T WP\Gamma)U \quad (24)$$

The conversion equation is as follows:

$$y = A^T\begin{bmatrix} \kappa(x_1, z) \\ \vdots \\ \kappa(x_N, z) \end{bmatrix} \quad (25)$$

3.3 NSK for speech emotion recognition

Based on our findings, we propose the following equations for the nonparametric speech emotion recognition. According to equation **Error! Reference source not found.** the modified $\lambda_l^{(i,j)}$ is named $\gamma_l^{(i,j)}$ (equation (26)) and define another matrix as $\gamma_l^{(i,j)}$, so that $\gamma_l^{un(i,j)}$ (equation (27)) is non-normalized weights of $\gamma_l^{(i,j)}$, then:

$$\gamma_l^{(i,i)} = \frac{\max\left(\gamma_l^{(i,i)}\right) - \gamma_l^{(i,i)}}{\sum_{t=1}^{N_i}\left\{\max\left(\gamma_t^{(i,i)}\right) - \gamma_t^{(i,i)}\right\}} \quad (26)$$

$$\gamma_l^{un\,(i,j)} = \Big[K_{ll}^{(i,i)} + (W^{(i,j)}K^{(j,j)}W^{(i,j)^T})_{ll} - 2(K^{(i,j)}W^{(i,j)^T})_{ll}\Big]^{-1/2} \qquad (27)$$

The modified $\gamma_l^{(i,j)}$ is namely a membership degree of each data to its class. In this case, the more the sample is far from the class center (boundary data), the more it gets weights.

We use the following equation (similar to Riley's distribution function) to reduce the effect of noisy data and eliminate them from the dataset. In this way, this equation is applied to the elements of matrix $\gamma_l^{(i,j)}$.

$$\gamma_l^{(i,j)} = \frac{\gamma_l^{(i,j)}}{0.606\sigma} e^{-\frac{\gamma_l^{(i,j)^2}}{2\sigma^2}} \qquad (28)$$

The σ value is determined such that 20% of each class's data is considered noisy. That is, according to the preceding equation, 20% of each class's data will be given less weight and will be deemed noisy data rather than border data. Now, we apply the equation to all the weights of $\gamma_l^{un(i,j)}$. Besides, the division to 0/606 is for normalization purposes.

$$\gamma_l^{un\,(i,j)} = \frac{\gamma_l^{un\,(i,j)}}{0.606} e^{-\frac{\gamma_l^{un\,(i,j)^2}}{2\sigma^2}} \qquad (29)$$

Then the final data weights for between-class and within-class values are as follows:

$$\lambda_l^{(i,i)} = 0.5(\gamma_l^{(i,i)} + \max_{\forall\, i\neq j}\gamma_l^{(i,j)})\,(\max_{\forall\, i\neq j}\gamma_l^{un(i,j)})^r \qquad (30)$$

$$r \in [0,\infty)$$

$$\lambda_l^{(i,j)} = 0.5(\gamma_l^{(i,i)} + \max_{\forall\, i\neq j}\gamma_l^{(i,j)})(\gamma_l^{un(i,j)})^r \qquad (31)$$

$$r \in [0,\infty)$$

In the following, the equations (30) and (31) are normalized and result in the equations (32) and (33). By powering one of the terms, the weight of the terms can be adjusted when multiplying them. The effect of $\gamma_l^{un(i,j)}$ is boosted by changing the power r in this equation. Therefore, the weights of $\gamma_l^{(i,j)}$ and $\gamma_l^{(i,i)}$ are substituted with the original formulation, which improves the results. The rest steps to get the conversion matrix are similar to the KNWFE algorithm.

$$\frac{\lambda_l^{(i,i)}}{\sum_l \sum_i \lambda_l^{(i,i)}} \rightarrow \lambda_l^{(i,i)} \qquad (32)$$

$$\frac{\lambda_l^{(i,j)}}{\sum_l \sum_i \sum_{j\neq i} \lambda_l^{(i,j)}} \rightarrow \lambda_l^{(i,j)} \qquad (33)$$

4 Experimental results

We use the confusion matrix to illustrate the results. The confusion matrix shows the number of correct classification and misclassifications for each class. If the number of classes is N, the confusion matrix is an N×N matrix that the main elements of its main diagonal represent the correct classifications in each class and the other elements represent misclassifications. Three criteria are extracted from this matrix, including the overall accuracy, the average accuracy, and the κ coefficient. Furthermore, the performance of the classifier is examined using the scatter plot. A Scatter plot shows the position of the data of each class relative to each other in a two-dimensional space (or two features) [6], [8]. Hughes curve is another evaluation criterion which we use in this paper. If X_{ii} is the correct number of classified data of each class, $i=1,2,\ldots,N$, and t denotes the total number of samples, the following equation presents the overall accuracy in percent [12]:

$$OA(\%) = \frac{\sum_{i=1}^{N} X_{ii}}{t} \times 100\% \qquad (34)$$

If the accuracy on each class is presented as equation (35), then the average accuracy in percent is calculated using equation (36) [12]:

$$AC_i = \frac{X_{ii}}{X_{i+}} \qquad (35)$$

$$AA(\%) = \frac{\sum_{i=1}^{N} AC_i}{N} \times 100\% \qquad (36)$$

where X_{i+} is the sum of the elements of the i^{th} row from the confusion matrix, which is equal to the total number of samples of class i. The κ coefficient is calculated from the following equation [12]:

$$\kappa = \frac{t\sum_{i=1}^{N} X_{ii} - \sum_{i=1}^{N} X_{i+}X_{+i}}{t^2 - \sum_{i=1}^{N} X_{i+}X_{+i}} \qquad (37)$$

where X_{+i} is the number of misclassified data of class i. The maximum value of κ is one. If the classifier does misclassify any data, the κ value is equal to one. In such a situation, only the main diagonal of the confusion matrix has non-zero values. The κ value is zero for a completely random classifier and is less than zero for a worse classifier [12].

4.1 Datasets

We used the Berlin database of emotional speech and the Persian ESD dataset to evaluate our proposed kernel method. In this database, ten actors (5 women and 5 men) play out emotions by reciting ten German phrases (5 short words and 5 lengthy sentences) that are commonly used in ordinary discussions and cover all of the interpretable emotions. The recordings were made in an anechoic room using high-quality recording equipment at the Technical University of Berlin's Department of Technical Acoustics. The database also contains 800 original sentences (10 (actors) × 7 (feelings) × 10 (sentences) + 100 (sentences from the second versions)). They evaluated the database in an audio test to accept the emotions and their naturalness. Emotional sentences with a naturality rate of more than 60% and the recognition rate of more than 80% remained in the database. About 500 sentences remained. Expert listeners selected these sentences. There are elements like anger, happiness, fear, sadness, boredom, disgust, and normal mode in the database. The total number of samples is 535, of which 302 and 233 samples are related to female and male speakers, respectively. The sample length is varied from 3 to 8 seconds, and the sampling frequency is 48 kHz that reaches 16 kHz after sample reduction [13].

The Persian ESD database, produced by Freie Universität Berlin, is the first comprehensive and official record of emotional speech in Persian. Two Persian-speaking actors (a man and a woman) participated in creating this series. The two actors perform 90 sentences in five emotional states of anger, joy, sadness, fear, disgust, and neutrality under certain conditions in three categories: congruent, incongruent, And baseline [14]. A group of 1126 native Persian speakers validated these sentences. The result is 472 vocal utterances with different emotional states. These signals were evaluated in an experiment by 34 native Persian speakers. In this process, 468 spoken utterances with a detection rate of greater than 71.42 percent were deemed legitimate. In this paper, we use 152 signals of the congruent section of this database, which includes 34, 30, 30, 30, and 28 utterances with emotions of anger, disgust, fear, happiness, and sadness, respectively.

We used hamming windowing with 20ms length and frame increment of 10ms (10ms overlap).

4.2 Test results

In this section, the proposed NSK algorithm is compared with the feature extraction algorithm of KNWFE and some other nonlinear methods based on Kernels such as GDA, KPCA, and linear features extraction algorithms, such as PCA and NWFE. This comparison is carried out on two introduced databases to recognize emotions. The experimental results on the Berlin and Persian databases are displayed in the next two subsections, which also examine the approaches' efficacy.

Experiments on Berlin Database

The Berlin database has seven senses, or in other words, seven classes. There are 127 utterances (audio file) for anger, 81 for boredom, 46 for disgust, 69 for fear, 71 for happiness, 62 for sadness, and 79 for the neutral state in this database. The first phase involves applying hamming windowing on the utterances with a length of 256 samples and an overlap of 128 samples in order to extract features. For this purpose, we use mostly-used features like MFCC, MFCC-DELTA, MFCC-DELT-DELTA, LPCC, pitch (including the maximum, the minimum, the range between the highest and lowest, mean, average, pitch deviation), formant, and energy. **Table 1** shows the number of extracted features for the Berlin and the Persian database. A total of 70 features were extracted from each utterance. It is important to note that adding some features (like Jitter and Shimer features in this test) does not affect the classification accuracy and overall accuracy of the recognition system of emotion, and it just increases the complexity of the system. However, depending on the research topic, more functional features are extracted.

Table 1 Extracted features for Persian and Berlin databases

Feature	Number of features
MFCC	12
Δ-MFCC	12
Δ-Δ-MFCC	12
Log-energy	6
LPCC	20
Pitch	6
formant	2

Table 2 shows the results of the Berlin database experiment with feature extraction algorithms and SVM and *k*-NN classifier.

Table 2 Evaluation of the NSK method and other methods on the Berlin database.

Criterion	Overall accuracy (%)		Average accuracy (%)		κ coefficient	
algorithm	SVM	*k*-NN	SVM	*k*-NN	SVM	*k*-NN

NSK	81.12	80.56	81.10	81.64	0.7775	0.7706
KNWFE	71.56	77.57	73.44	78.71	0.6675	0.7352
KPCA	71.96	78.69	73.14	79.23	0.6693	0.7490
GDA	78.50	78.86	79.51	80.62	0.7468	0.7516
PCA	77.38	77.75	78.79	78.91	0.7341	0.7385
NWFE	62.42	62.99	66.43	66.95	0.5612	0.5667

It is clear from the results of **Table 2** that the NSK algorithm outperforms the other feature extraction algorithms. It is trivial to mention that the performance of linear feature extraction algorithms may also be close to nonlinear methods for a dataset if the classes of the dataset are relatively linearly separable, and the assumption that the data is Gaussian is correct for that data set. We randomly select 40 samples from each class or emotion to teach the SVM classifier, and repeat the operation five times and average the results to generalize the outcome. Cross-validation is then used to obtain the best value for the parameters of each algorithm. The variation of σ in the Gaussian kernel for the nonlinear feature extraction algorithm is {0.02, 0.2, 2, 20, …, 2000}, and it is {0.1, 0.2, 2, …, 20} for the other methods, the range of C value for all methods is {0.1, 1, 10, 100, 10^3, …, 10^6}, the number of the neighbors of k-NN classifier is {1, 3, 5, 7, 9, 11}, and the range of Minkowski's distance rank is {2, 3, 4, 5}. **Table 3** presents the best-adjusted parameters for the feature extraction algorithm and Berlin database.

Table 3 Best empirical parameters for feature extraction algorithms on the Berlin database.

Method	NSK	KNWFE	KPCA	GDA	PCA	NWFE
Kernel	G	G	G	G	-	-
σ	50	5.5	9	100	-	-
SVM kernel	G	G	G	G	G	G
SVM kernel width	5	2.2	0.68	0.7	11	0.1
C	10	10000	100	100	100	1000
No. of neighbours of k-NN	7	5	3	3	9	5
Distance Order	M 3	M 3	M 3	M 2	M 2	M 3

* G= Gaussian, M=Minkowski

Fig. 1 and **Fig. 2** present the recognition rate of the NSK method in comparison with the k-NN and SVM classifiers.

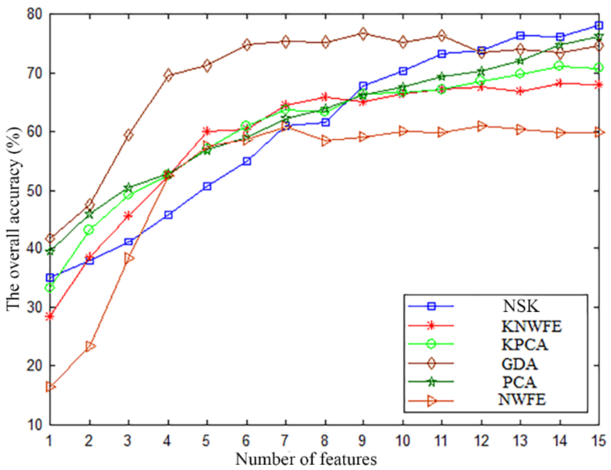

Fig. 1 A comparison of SVM classification using NSK and other features on the Berlin database.

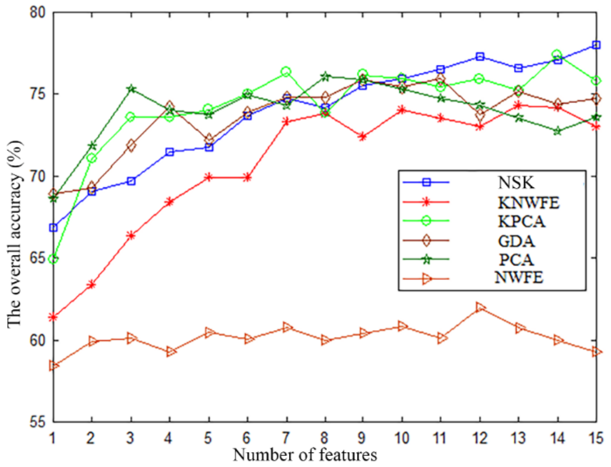

Fig. 2 A comparison of k-NN classification using NSK and other features on the Berlin database.

The increased performance of the NSK kernel over the others is shown in **Fig. 1** and **Fig. 2**. As can be seen, the number of features increases the separability of classes and classification accuracy. We have specifically chosen to display only the first 15 features in Figures 1 and 2 as the performance differences between various methods are more pronounced in these features. The features were selected in the order shown in Table 1. According to the Hughes phenomenon, the classification accuracy falls as the number of features grows. **Fig. 1** and **Fig. 2** also depict the NSK method presents more classification accuracy than the other methods. **Table 4** and **Table 5** show the confusion matrix, and **Table 6** shows the classification accuracy percentage of each class for the NSK method and other methods using an SVM classifier.

Table 4 Confusion matrix for the NSK features and SVM classifier on the Berlin database.

Emotion	A	B	D	F	H	S	N
A	101	0	1	3	21	0	1
B	0	63	3	5	1	2	7
D	1	0	44	0	1	0	0
F	3	1	2	55	4	3	1
H	6	1	1	7	56	0	0
S	0	1	0	1	0	60	0
N	0	14	4	4	2	0	55

* A=Anger, B=Boredom, D=Disgust, F=Fear, H=Happiness, S=Sadness, N=Neutral.

Table 5 Confusion matrix for the KNWFE features and SVM classifier on the Berlin database.

Emotion	A	B	D	F	H	S	N
A	85	2	9	20	9	2	0
B	0	53	7	9	1	6	5
D	2	0	42	2	0	0	0
F	2	2	4	55	3	2	1
H	7	0	10	8	45	0	1
S	0	0	7	2	1	49	3
N	0	6	6	10	1	2	54

* A=Anger, B=Boredom, D=Disgust, F=Fear, H=Happiness, S=Sadness, N=Neutral.

Table 6 Classification accuracy for different feature reduction methods using SVM on the Berlin database in terms of percentage.

Feature reduction algorithm	A	B	D	F	H	S	N
NSK	79.52	77.77	95.65	79.71	78.87	96.77	69.62
KNWFE	66.92	65.43	91.30	79.71	76.05	91.93	68.35
KPCA	73.22	56.79	89.13	84.05	69.01	87.09	62.02
GDA	79.52	75.30	91.30	75.36	76.05	91.93	67.08
PCA	77.95	71.60	95.65	72.46	80.28	90.32	63.29
NWFE	44.09	60.49	91.30	75.36	61.97	74.19	56.96

* A=Anger, B=Boredom, D=Disgust, F=Fear, H=Happiness, S=Sadness, N=Neutral.

According to Table 6, the feature extraction algorithms almost have had a good performance in identifying emotions of disgust. Since it creates features in speech signals, it has a border that is distinguishable from other emotions. The happiness, fear, and anger emotions are almost close to each other this is why feature extraction algorithms fail to identify emotion boundaries. **Table 6** also shows that the NSK algorithm has a better performance in contrary to others to separate these three emotions. Moreover, boredom, sadness, and neutral emotions are almost identical. In this case, the suggested method also has a better performance in separating these emotions. **Table 4** and **Table 5**, which are confusion matrices of the NSK and KNWFE algorithms, show that the NSK is more effective than the other algorithms in identifying anger from happiness. These two emotions have high excitement and are different just in the amount. **Table 7** to **Table 9** show the emotion recognition results for the k-NN classifier.

Table 7 Confusion matrix for the NSK features and k-NN classifier on the Berlin database.

Emotion	A	B	D	F	H	S	N
A	105	0	0	8	13	0	1
B	0	60	1	6	1	3	10
D	0	0	42	2	1	0	1
F	4	2	1	57	3	1	1
H	6	0	3	6	54	0	2
S	0	1	1	1	0	59	0
N	0	10	2	10	2	1	54

* A=Anger, B=Boredom, D=Disgust, F=Fear, H=Happiness, S=Sadness, N=Neutral.

Table 8 Confusion matrix for the KNWFE features and k-NN classifier on the Berlin database.

Emotion	A	B	D	F	H	S	N
A	101	1	2	13	10	0	0
B	2	54	3	4	1	4	13
D	0	1	42	0	1	0	2
F	3	2	2	55	4	0	3
H	9	1	1	4	54	0	2
S	0	4	0	1	0	57	0
N	0	12	0	8	2	5	52

* A=Anger, B=Boredom, D=Disgust, F=Fear, H=Happiness, S=Sadness, N=Neutral.

Table 9 Classification accuracy for different feature reduction methods using k-NN on the Berlin database in terms of percentage.

Feature reduction algorithm	A	B	D	F	H	S	N
NSK	82.67	74.07	91.30	82.60	76.05	95.16	68.35
KNWFE	79.52	66.66	91.30	79.71	76.05	91.93	65.81
KPCA	79.52	67.90	97.82	81.15	76.05	96.77	67.08
GDA	78.74	60.49	100	72.46	80.28	95.16	77.21
PCA	83.46	64.19	100	66.66	70.42	96.77	70.88
NWFE	45.66	59.25	93.47	62.31	59.15	85.48	63.29

* A=Anger, B=Boredom, D=Disgust, F=Fear, H=Happiness, S=Sadness, N=Neutral.

The NSK method and KNWFE have the same performance in identifying disgust from happiness emotion using the k-NN classifier and have the best performance in identifying boredom and fear emotion categories. These results also suggest that the proposed method is not the most appropriate (nor the worst) using the k-NN for the Berlin database. **Fig. 3** and **Fig. 4** show the scatter plot of the NSK and KNWFE algorithms for the first and second features. We used three emotions of anger, boredom, and disgust to draw this map.

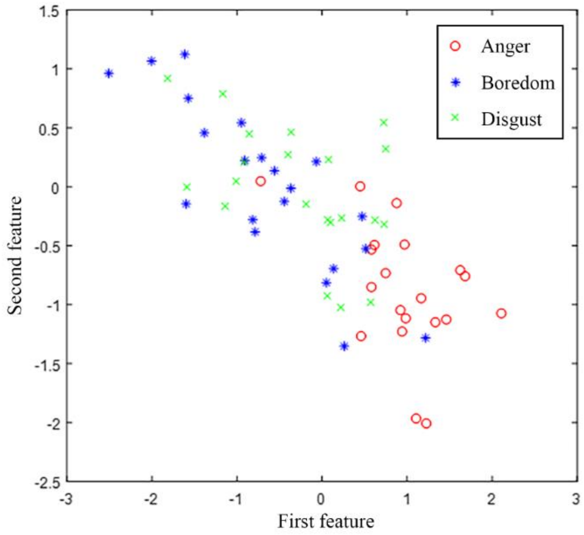

Fig. 3 Distribution of NSK features for Berlin database training samples.

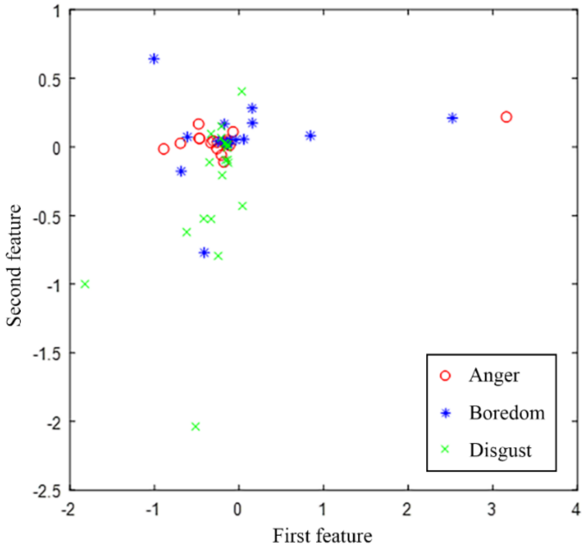

Fig. 4 Distribution of KNWFE features for Berlin database training samples.

The scatter plot shows the situation of classes to each other. As it is clear from **Fig. 3** and **Fig. 4**, the NSK algorithm has better separability in the scatter plot than the KNWFE method.

Experiments on the Persian Database

The Persian database has five emotions, such as anger, disgust, fear, happiness, and sadness. Like the case of the Berlin database, we used the features of **Table 1** for this database. **Table 10** shows the results of the test on the Persian database using the k-NN and SVM classifiers. We randomly selected 20 samples from each class or emotion to train the SVM classifier, and repeat the operation five times and average the results to generalize the outcome. Cross-evaluation is also used in this experiment. The variation of σ in the Gaussian kernel for the nonlinear feature extraction algorithm is {0.02, 0.2, 2, 20, …, 200}, and {0.1, 0.2, 2, …, 20} for the other methods, the range of C value for all methods is {0.1, 1, 10, 100, 10^3, …, 10^6}, the number of the neighbors of k-NN classifier is {1, 3, 5, 7, 9, 11 and the range of Minkowski's distance rank is {2, 3, 4, 5}. **Table 11** presents the best-adjusted parameters for the feature extraction algorithm and Persian database.

Table 10 Evaluation of the NSK method and other methods on the Persian database.

Criterion	Overall accuracy (%)		Average accuracy (%)		κ coefficient	
Feature reduction algorithm	SVM	*k*-NN	SVM	*k*-NN	SVM	*k*-NN
NSK	96.71	96.05	96.57	95.90	0.9588	0.9506
KNWFE	90.78	91.44	91.24	91.37	0.8849	0.7352
KPCA	88.15	92.10	88.17	92.34	0.8519	0.9013
GDA	91.44	90.13	91.55	90.23	0.8929	0.8765
PCA	92.76	91.44	92.85	91.47	0.9095	0.8930
NWFE	83.55	85.52	83.63	85.94	0.7944	0.8191

Table 11 Best empirical parameters for feature extraction algorithms on the Persian database.

Method	NSK	KNWFE	KPCA	GDA	PCA	NWFE
Kernel	G	G	G	G	-	-
σ	30	4.5	10	200	-	-
SVM kernel	G	G	G	G	G	G
SVM kernel width	5	2.4	0.6	0.75	12	0.11
C	10	10000	100	100	100	1000
No. of neighbors of *k*-NN	7	7	7	5	7	7
Distance	M	M	M	M	M	M
Order	3	3	3	2	2	3

* G= Gaussian, M= Minkowski.

Table 10 shows that the NSK method has the best performance compared to the baseline methods. An equal number of data in each class referred known as "parallelization of class data," aids the algorithm's performance. The number of data in each class of the Persian database is almost equal. This leads to the improvement of the performance of feature extraction algorithms contrary to the Berlin database. In many studies, researchers parallelize all class data before the feature extraction step. In this study, the original form of the database is used. However, we use databases in the same way as the original and do not use class data parallelization methods to test the performance of the proposed method in a more real condition.

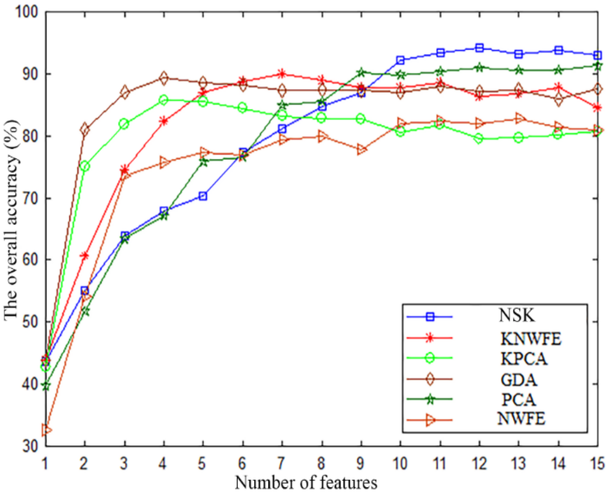

Fig. 5 A comparison of SVM classification using NSK and other features on the Persian database.

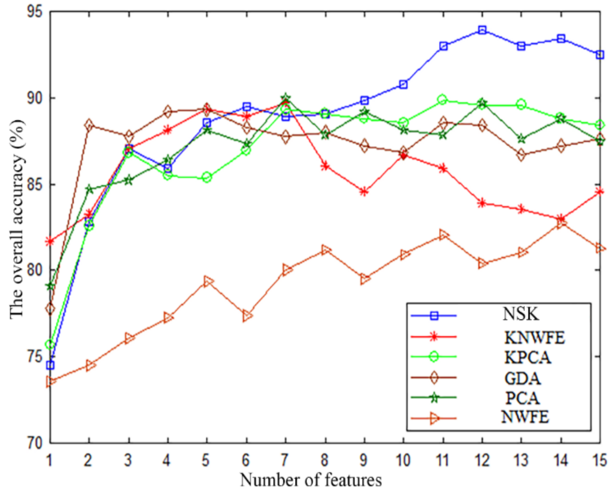

Fig. 6 A comparison of k-NN classification using NSK and other features on the Persian database.

Fig. 5 and **Fig. 6** show the NSK method in comparison to other algorithms using SVM and k-NN classifier, respectively. **Table 12** and **Table 13** show the confusion matrices, and **Table 14** shows the classification accuracy of each class in percentage for the NSK and KNWFE methods.

Table 12 Confusion matrix for the NSK features and SVM classifier on the Persian database.

Emotion	A	D	F	H	S
A	34	0	0	0	0
D	0	30	0	0	0
F	1	0	29	0	0
H	0	0	1	28	1
S	1	0	0	1	26

* A=Anger, D=Disgust, F=Fear, H=Happiness, S=Sadness.

Table 13 Confusion matrix for the KNWFE features and SVM classifier on the Persian database.

Emotion	A	D	F	H	S
A	34	0	0	0	0
D	0	30	0	0	0
F	1	0	29	0	0
H	0	0	1	28	1
S	1	0	0	1	26

* A=Anger, D=Disgust, F=Fear, H=Happiness, S=Sadness.

Table 14 Classification accuracy for different feature reduction methods using SVM on the Persian database in terms of percentage.

Feature reduction algorithm	A	D	F	H	S
NSK	100	100	96.66	93.33	92.85
KNWFE	76.47	100	100	83.33	96.42
KPCA	88.23	93.33	90	80	89.28
GDA	88.23	90	96.66	90	92.85
PCA	91.17	93.33	90	93.33	96.42
NWFE	85.29	73.33	93.33	73.33	92.85

* A=Anger, D=Disgust, F=Fear, H=Happiness, S=Sadness.

Table 15 Confusion matrix for the NSK features and k-NN classifier on the Persian database.

Emotion	A	D	F	H	S
A	34	0	0	0	0
D	0	29	0	0	1
F	0	0	29	0	1
H	2	0	0	28	0
S	1	0	0	1	26

* A=Anger, D=Disgust, F=Fear, H=Happiness, S=Sadness.

Table 16 Confusion matrix for the KNWFE features and k-NN classifier on the Persian database.

Emotion	A	D	F	H	S
A	31	0	2	1	0
D	0	28	0	1	1
F	0	0	30	0	0
H	1	0	2	26	1
S	0	0	4	0	24

* A=Anger, D=Disgust, F=Fear, H=Happiness, S=Sadness.

Table 17 Classification accuracy for different feature reduction methods using KNN on the Persian database in terms of percentage.

Feature reduction algorithm	A	D	F	H	S
NSK	100	96.66	96.66	93.33	92.85
KNWFE	91.17	93.33	100	86.66	85.71
KPCA	85.29	93.33	96.66	90	96.42
GDA	91.17	93.33	83.33	83.33	100
PCA	91.17	100	83.33	90	92.85
NWFE	73.52	93.33	86.66	83.33	92.85

* A=Anger, D=Disgust, F=Fear, H=Happiness, S=Sadness.

As it is clear from **Table 14**, the NSK method shows better performance, contrary to other methods. The NSK performed better in identifying close emotions like anger and happiness compared to others. **Fig. 7** and **Fig. 8** show the scatter plot of the NSK and KNWFE for the first and second features. Three emotions of anger, disgust and fear were used in this plot. **Table 15** to **Table 17** show the confusion matrix of the NSK and KNWFE and classification accuracy by each emotion for the NSK and other methods for the k-NN classifier, respectively.

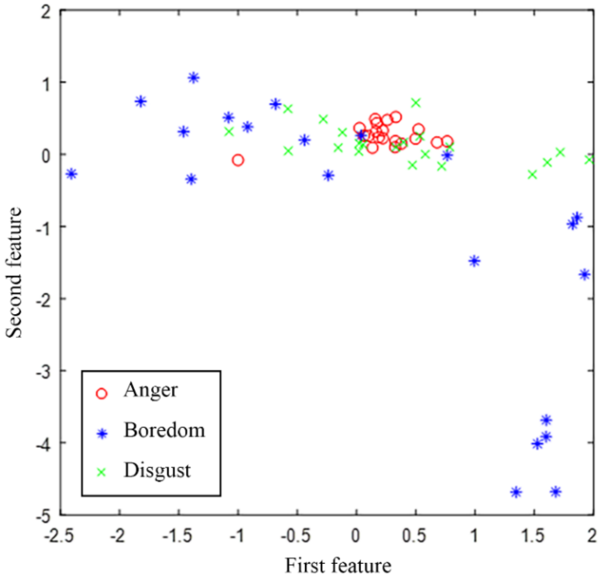

Fig. 7 Distribution of NSK features for Persian database training samples.

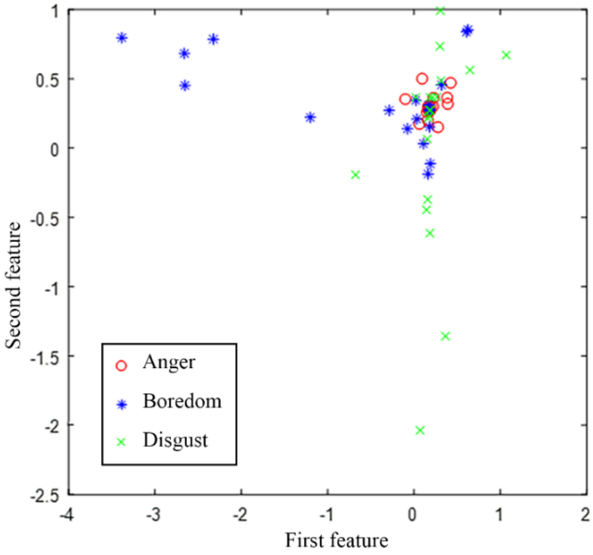

Fig. 8 Distribution of KNWFE features for Persian database training samples.

5 Conclusion

In this paper, a nonlinear kernel for speech signal is proposed and implemented, which provides high accuracy in recognizing the emotions from the speech signal. By reviewing and implementing various methods, we found flaws in the KNWFE method and tried to fix them properly. We revised the relationships of the KNWFE method and modified some equations. The improvement of equations led to adding some free parameters to the problem and consequently increased the computational complexity. With this additional complexity, it is time-consuming to find these free parameters using cross-validation. The simulation time is 63 seconds for the KNWFE method, and 78 seconds for the NSK on the Berlin database, 42 seconds for the KNWFE, and 57 seconds for the NSK on the Persian dataset. A computer with hardware specifications: CPU: Core i5 3210M and RAM: 6GB is used to simulate this data set. The proposed method (NSK) is implemented with the emotional speech datasets and evaluated and analyzed the results. Eventually, the experiments show that the proposed method has a promising performance. The results show an overall accuracy of 81.12% for the Berlin database and 96.71% for the Persian database, which have 10% and 6% improvement, respectively, compared to the conventional KNWFE method. As future research, the proposed algorithm can be evaluated in all speech researches that take advantage of a kernel.

Acknowledgment

This research was supported by the University of

Jiroft under the grant No 5812-02-3. [15]

Reference

[1] K. S. Rao and S. G. Koolagudi, “Robust emotion recognition using spectral and prosodic features,” *Springer Science \& Business Media*, 2013.

[2] J. de Lope and M. Graña, “An ongoing review of speech emotion recognition,” *Neurocomputing*, vol. 528, pp. 1–11, 2023, doi: https://doi.org/10.1016/j.neucom.2023.01.002.

[3] B.-C. Kuo and D. A. Landgrebe, “Nonparametric weighted feature extraction for classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 5, pp. 1096–1105, 2004.

[4] F. Daneshfar, S. J. Kabudian, and A. Neekabadi, “Speech emotion recognition using hybrid spectral-prosodic features of speech signal/glottal waveform, metaheuristic-based dimensionality reduction, and Gaussian elliptical basis function network classifier,” *Applied Acoustics*, vol. 166, p. 107360, 2020, doi: https://doi.org/10.1016/j.apacoust.2020.107360.

[5] S. Gupta, M. S. Fahad, and A. Deepak, “Pitch-synchronous single frequency filtering spectrogram for speech emotion recognition,” *Multimedia Tools and Applications*, vol. 79, pp. 23347–23365, 2020, doi: 10.1007/s11042-020-09068-1.

[6] H. Guan, Z. Liu, L. Wang, J. Dang, and R. Yu, “Speech Emotion Recognition Considering Local Dynamic Features,” in *Studies on Speech Production*, Y. N. Fang Q., Dang J., Perrier P., Wei J., Wang L., Ed. Lecture Notes in Computer Science, vol 10733. Springer, Cham., 2018.

[7] B. Schuller and G. Rigoll, “Timing Levels in Segment-Based Speech Emotion Recognition,” in *INTERSPEECH*, 2006, pp. 1818–1821.

[8] K. Wang et al., *Speech Emotion Recognition Using Fourier Parameters*, vol. 6, no. 1, pp. 69–75, 2015.

[9] R. V Darekar, M. Chavan, S. Sharanyaa, and N. M. Ranjan, “A hybrid meta-heuristic ensemble-based classification technique speech emotion recognition,” *Advances in Engineering Software*, vol. 180, p. 103412, 2023, doi: https://doi.org/10.1016/j.advengsoft.2023.103412.

[10] X. Xu et al., “A two-dimensional framework of multiple kernel subspace learning for recognizing emotion in speech,” *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 25, no. 7, pp. 1436–1449, 2017, doi: 10.1109/TASLP.2017.2694704.

[11] B. C. Kuo, C. H. Li, and J. M. Yang, “Kernel nonparametric weighted feature extraction for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 4, pp. 1139–1155, 2009, doi: 10.1109/TGRS.2008.2008308.

[12] D. A. Landgrebe, *Signal theory methods in multispectral remote sensing*, vol. 29. John Wiley & Sons, 2005.

[13] F. Burkhardt et al., *A Database of German Emotional Speech*, pp. 1517–1520, 2005.

[14] N. Keshtiari, M. Kuhlmann, M. Eslami, and G. Klann-Delius, “Recognizing emotional speech in Persian: A validated database of Persian emotional speech (Persian ESD),” Behav. Res. Methods, vol. 47, pp. 275–294, 2015.

O**orappan G Murugan** Mohammad Hasheminejad may be an accomplished Assistant Professor at the University of Jiroft in the Kerman province, Iran, specializing in Telecommunication Engineering. Having earned a PhD in 2016, from the University of Birjand, in the field of Telecommunication Engineering, he has since been an active researcher with a keen interest in data science, artificial intelligence, and robotics. Proficient in MATLAB and Python programming, Dr. Mohammad Hasheminejad has made significant contributions to the field and has been recognized with three awards as one of the best researchers in 2023 at the University of Jiroft. With a strong publication record and a dedication to advancing the frontiers of technology.