Sürdürülebilir Mühendislik Uygulamaları veTeknolojik Gelişmeler Dergisi
Journal of Sustainable Engineering Applications and Technological Developments
2023, 6(2): 235-249

Makine Öğrenimi ve Hibrit Altuzay Sınıflandırıcılar için Yalıtık Kelime Tanıma Performanslarının Karşılaştırılması

A Comparison of Isolated Word Recognition Performances for Machine Learning and Hybrid Subspace Classifiers

Serkan KESER[1*]

[1*] Kırşehir Ahi Evran University, Faculty of Engineering and Architecture, Kırşehir, Türkiye

ÖZET

Konuşma tanıma çalışmalarında tanıma oranlarını etkileyen temel faktörlerden biri çevresel arka plan gürültüsüdür. Bu çalışmada, konuşmacıdan bağımsız izole kelime tanıma işlemini gerçekleştirmek için farklı gürültü türlerini içeren bir konuşma veritabanı kullanılmıştır. Böylece gürültülü konuşma sinyallerinin sınıflandırıcıların tanıma performansı üzerindeki etkilerini anlamak mümkün olacaktır. Çalışmada K-En Yakın Komşular (KNN), Fisher Doğrusal Diskriminant Analizi-KNN (FLDA-KNN), Ayrımcı Ortak Vektör Yaklaşımı (DCVA), Destek Vektör Makineleri (SVM), Evrişimsel Sinir Ağı (CNN) ve Tekrarlayan Sinir Ağı kullanılmıştır. Sınıflandırıcı olarak Uzun Kısa Süreli Bellek (RNN-LSTM) kullanıldı. Özellik vektörleri olarak MFCC ve PLP katsayıları kullanıldı. DCVA sınıflandırıcısı, literatürde ilk kez izole edilmiş kelime tanıma açısından derinlemesine test edilmiştir. Tanıma işlemi KNN, FLDA-KNN ve DCVA sınıflandırıcıları için çeşitli mesafe ölçütleri kullanılarak gerçekleştirilmiştir. Ayrıca, yeni $(DCVA)_{PCA}$ ve $(FLDA\text{-}KNN)_{PCA}$ sınıflandırıcıları, Temel Bileşen Analizi (PCA) kullanılarak hibrit algoritmalar olarak tasarlanmış ve DCVA ve FLDA-KNN sınıflandırıcılarından daha iyi tanıma sonuçları elde edilmiştir. En yüksek tanınma oranı deneysel çalışmalarda RNN-LSTM ile %93,22 bulunmuştur. Diğer sınıflandırıcılar için ise en yüksek tanınma oranları sırasıyla CNN, KNN, DCVA, $(DCVA)_{PCA}$, SVM, FLDA-KNN ve $(FLDA\text{-}KNN)_{PCA}$'nın %87,56, %86,51, %74,23, %79, %77,78, %71,37 ve %84,90'dir.

Anahtar Kelimeler: Gürültülü Konuşma Sinyalleri, Hibrit Alt Uzay Sınıflandırıcıları, Makine Öğrenimi Sınıflandırıcıları, PLP, MFCC

ABSTRACT

One of the essential factors affecting recognition rates in speech recognition studies is environmental background noise. This study used a speech database containing different noise types to perform speaker-independent isolated word recognition. Thus, it will be possible to understand the effects of speech signals having noise on the recognition performance of classifiers. In the study, K-Nearest Neighbors (KNN), Fisher Linear Discriminant Analysis-KNN (FLDA-KNN), Discriminative Common Vector Approach (DCVA), Support Vector Machines (SVM), Convolutional Neural Network (CNN), and Recurrent Neural Network with Long Short-Term Memory (RNN-LSTM) were used as classifiers. MFCC and PLP coefficients were used as feature vectors. The DCVA classifier has been deeply tested for isolated word recognition for the first time in the literature. The recognition process was carried out using various distance measures for the KNN, FLDA-KNN, and DCVA classifiers. In addition, new $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ classifiers were designed as hybrid algorithms using Principle Component Analysis (PCA), and better recognition results were obtained from those of DCVA and FLDA-KNN classifiers. The highest recognition rate of RNN-LSTM was 93.22% in experimental studies. For the other classifiers, the highest recognition rates of the CNN, KNN, DCVA, $(DCVA)_{PCA}$, SVM, FLDA-KNN, and $(FLDA\text{-}KNN)_{PCA}$ were 87.56%, 86.51%, 74.23%, 79%, 77.78%, 71.37% and 84.90%, respectively.

Keywords: Noisy Speech Signals, Hybrid Subspace Classifiers, Machine Learning Classifiers, PLP, MFCC

Başvuru: 07.08.2023 Son Revizyon: 24.09.2023 Kabul: 04.10.2023
Doi: 10.51764/smutgd.1338977

[1*]Sorumlu yazar: Kırşehir Ahi Evran Üniversitesi, Mühendislik Mimarlık Fakültesi, Bağbaşı kampüsü, Kırşehir, Türkiye; E-mail: skeser@ahievran.edu.tr; ORCID: 0000-0001-8435-0507

1. INTRODUCTION

Speech recognition systems are used in many different fields as smart homes where domestic devices are controlled by voice commands, control of robots and vehicles, interactive voice response systems, speech dictation, speech emotion recognition, and speaker recognition (Filho & Moir, 2010; Anggraeni, 2018; Soujanya and Kumar, 2010; Furui et al., 2004; Beigi, 2011; Lalitha, 2015; Akyazi et al., 2019). Speech recognition can be performed speaker-dependent or speaker-independent for a speech database. The size of the database and speaker dependency factors also have an essential effect on the recognition rates of the classifiers.

The Dynamic Time Warping (DWT) algorithm, one of the most used speech recognition classifiers in the literature, is a matching method used in the similarity measurement of time series. On the other hand, the overall recognition rate of DWT is lower than those of different classifiers (Permanasari et al., 2020). The Hidden Markov Model (HMM) is a classifier that uses a language model and gives high recognition rates, especially in real-time and speaker-independent recognition (Palaz et al., 2019; Muhammad et al., 2020; Tokuda et al., 2000). Recurrent Neural Networks (RNN), using Long-Short-Term Memory (LSTM) architecture is a deep learning algorithm that is widely used in speech recognition today and is known to give satisfying results (Sak et al., 2014; Dokuz and Tüfekci, 2020). Another deep learning algorithm used in voice recognition is Convolutional Neural Networks (CNN). Using the CNN algorithm as a classifier, high recognition rates can be obtained in speech recognition (Guleti et al., 2020; Dokuz and Tüfekci, 2020).

Another family of classifiers used in speech recognition is subspace classifiers. Basic subspace classifiers used in image or speech recognition in the literature are known as Fisher Linear Discrimination Analysis (FLDA), Class Featuring Information Compression (CLAFIC), and Common Vector Approach (CVA) (Keser and Edizkan, 2009; Yavuz et al., 2006; Gunal and Edizkan, 2008). The CVA, a subspace classification method, gives high recognition rates in limited isolated word recognition applications (Gülmezoglu, 1999; Gunal and Edizkan, 2008; Keser and Edizkan, 2009). In addition, CVA can work faster than many classifiers mentioned above because it uses one vector representing each class, making it attractive for real-time speech or image recognition applications. First, the Discriminative Common Vector Approach (DCVA), based on the CVA method and used mainly in face recognition applications, was introduced by Çevikalp (2005). The DCVA approach can give better results in face recognition applications than other subspace methods such as Eigenface and FLDA classifiers (Çevikalp, 2005). According to the CVA, the advantage of the DCVA is that it uses feature vectors of size one less than the number of classes for each class. Also, the DCVA has a lower computation time than the Eigenface and FDAA subspace methods, just like the CVA. The FLDA, another essential subspace method, is a classifier generally used in face recognition studies and is based on the Linear Discrimination Analysis (LDA) method (Kolossa et al., 2013; Song et al., 2014). However, it is also used in speech recognition studies (Srisuwan et al., 2018, Sivaram et al., 2010).

Background noise added to the microphone apart from the human speech is essential for real speech recognition systems. Many studies are used to recognize speech signals containing noise in the literature. One of these has been introduced to investigate noise robustness (Seltzer et al., 2013). This work is a new acoustic model based on deep neural networks (DNN) and HMM. Another study is based on the very deep convolutional residual network (VDCRN). This paper proposes a more advanced model referred to as the VDCRN (Tan et al., 2018), and speech recognition was tested in noisy environments. Finally, Sumit et al. proposed an end-to-end deep learning approach leveraging current progress in the automatic speech recognition system to recognize continuous Bangla speech in noisy environments.

Therefore, examining the classifier's performance in speech recognition for noise-containing speech signals will be essential. The study performed a classification process using a speech database containing various noises for speaker-independent isolated word recognition. This work also used machine learning algorithms, like the CNN, KNN, SVM, RNN-LSTM, and the DCVA and FLDA subspaces classifiers. Furthermore, algorithms based on the different distance measures (Euclidean, Correlation, Cityblock, Spearman) were applied in the DCVA, FLDA-KNN, and KNN for more detailed investigations. In addition, for the first time in the literature for speech recognition, the DCVA was deeply tested using sufficient data states and different distance measures. The number of classes used in the study is 18, and 150 speech signals were used by taking speech signals ranging from 1 to 5 from people randomly selected in the speech database for each class. Perceptual Linear Prediction (PLP) and MFCC coefficients were found for each frame whose length is 20 ms, and the overlap between consecutive frames is 50%. In addition, the study used 13 or 39-dimensional MFCC and PLP coefficients for each frame. The 39 coefficients are performed by combining 13 MFCC (or PLP),13 delta, and 13 delta-delta coefficients. In some of the test processes, the

dimensions of these feature vectors were reduced using PCA, and then algorithms such as DCVA or FLDA-KNN were used, while other tests were carried out without reducing the size with PCA. Finally, the DCVA and FLDA-KNN algorithms used PCA were named $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$. In the testing phase, 3-fold cross-validation was applied for all classifiers, and the average recognition performances were obtained using these classifiers. Finally, the RNN-LSTM gave the best recognition performance in experimental studies (93.21%).

2. THE PROPOSED CLASSIFIER ALGORITHMS

The CNN, KNN, RNN-LSTM, SVM, FLDA-KNN, $(FLDA\text{-}KNN)_{PCA}$, DCVA, and $(DCVA)_{PCA}$ algorithms were preferred in the study. These classifiers are frequently used for speech recognition in the literature except for the DCVA.

2.1 The Proposed RNN-LSTM Algorithm

An LSTM network is a Recurrent Neural Network (RNN) type that can learn long-term dependencies between time steps of sequence data (Tan and Wang, 2018). An RNN is a deep-learning network structure that uses past information to improve the network's performance on current and future inputs. The used RNN-LSTM architecture is shown in Figure 1. This diagram illustrates the architecture of a simple LSTM network for classification.

Figure 1. The proposed RNN-LSTM algorithm structure

The network starts with a sequence input layer followed by an LSTM layer. The network ends with a fully connected layer, a SoftMax layer, and a classification output layer to predict class labels. Fully connected layers connect every neuron in one layer to every neuron in the next layer (Tan and Wang, 2018). The SoftMax layer includes a SoftMax function that turns a vector of N real values that sum to 1. On the other hand, the SoftMax layer assigs real probabilities to each class. At the output, the signal is assigned to the relevant class according to the values from the SoftMax layer. The LSTM has the forget, input, and output states, which help the network reduce the long-term dependency on data. Forget state removes redundant data while the input state processes new data (Garcia et al., 2020). The block diagram of the LSTM cell is shown in Figure 1 (Garcia et al., 2020).

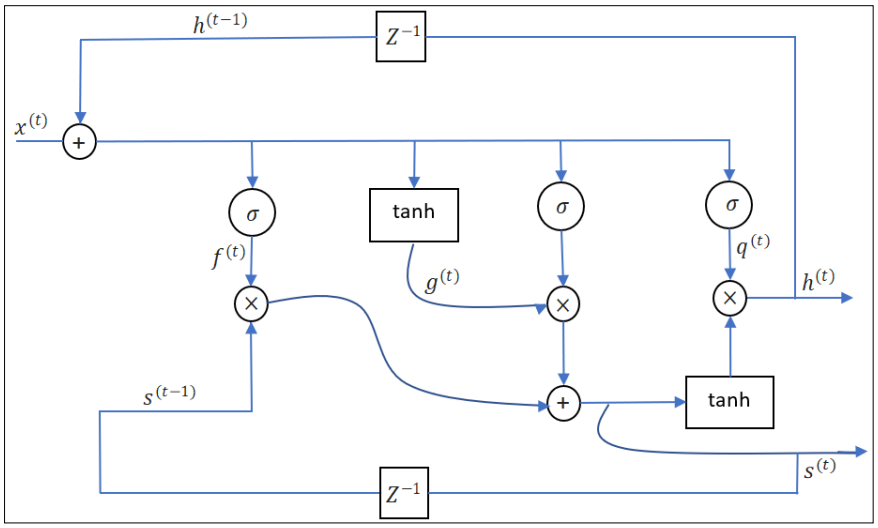

Figure 2. Block diagram of one cell of a long-short term memory architecture.

In Figure 2, The forget state controls the state parameter s(t) via a sigmoid function s. f (t) is the forget vector, x(t) and h(t-1) are the input and previous output, respectively. g(t) is the next candidate for the cell state.

2.2. The Proposed CNN Algorithm

The proposed network consists of only five convolution layers with filters. Each convolution layer has batch normalization, ReLU, and maximum pooling layers. After the five stages of convolution, the network has a fully connected layer, a SoftMax layer, and finally, the classification layer. The Adam optimizer is used to train the network. The feature vectors obtained from the speech signals are used as the input data. The used block diagram of the architecture is shown in Figure 3.

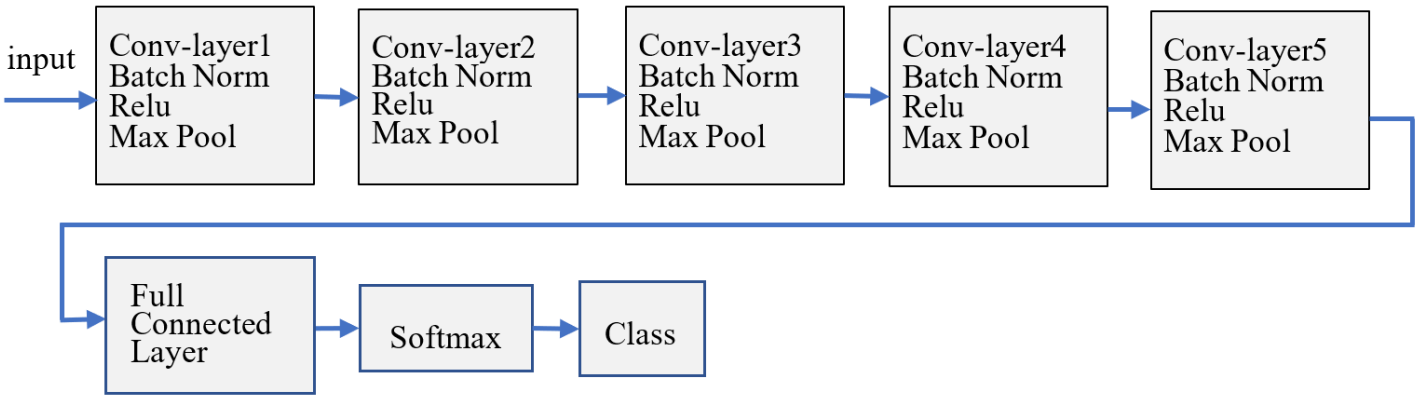

Figure 3. CNN architecture block diagram

2.3. The Proposed SVM Algorithm

The SVM is a machine learning algorithm mostly used for classification, which can be performed for two or more classes using SVM (Miao et al.,2018). Linear, polynomial, and radial basis kernel functions are generally used for classification. These kernels provide a more accurate classification by assigning the feature vectors of the classes to higher dimensions. With equations, the radial basis function (RBF), Linear, and Polynomial kernels are given below. The RBF kernel is one of the most widely used kernels due to its similarity to the Gaussian distribution and can be represented as $K(x_1, x_2)$ given by

$$K(x_1, x_2) = e^{-a\|x_1 - x_2\|^2} \tag{1}$$

where the variable a varies from 0 to 1, and x_1 and x_2 are vectors in the input space. The following equation represents the linear kernel:

$$K(x_1, x_2) =< \varphi(x_1), \varphi(x_2) > \tag{2}$$

where φ is the function that maps the x's to linear space, the polynomial kernel is a kernel function commonly used with support vector machines (SVMs). For degree-d polynomials, the polynomial kernel is defined as (Zhang and Gales, 2012)

$$K(x_1, x_2) = (1 + {x_1}^{\top} x_2)^d \tag{3}$$

2.4. The Proposed KNN Algorithm

The KNN is a machine-learning algorithm that can solve classification and regression problems (Soucy and Mineau, 2001). This algorithm uses the number of neighbors and distance measures such as Minkowski, Euclidean, Cityblock, Spearman, Correlation, and Chebyshev. The test signal is assigned to the most appropriate class calculated using the nearest neighbor algorithm (Song et al., 2007). The study used Euclidean, Cityblock, Spearman, and Correlation distance measures for the proposed KNN algorithm.

2.5. The Proposed DCVA Algorithm

Its low computational complexity is the most important feature distinguishing the DCVA from other subspace methods. The main reason for this advantage is that it uses a single discriminative common vector representing each class (Gulmezoglu et al., 1999). The block diagram of the proposed DCVA algorithm is given in Figure 4 below. In Figure 4, the letters SDM stand for the Selected Distance Measure.

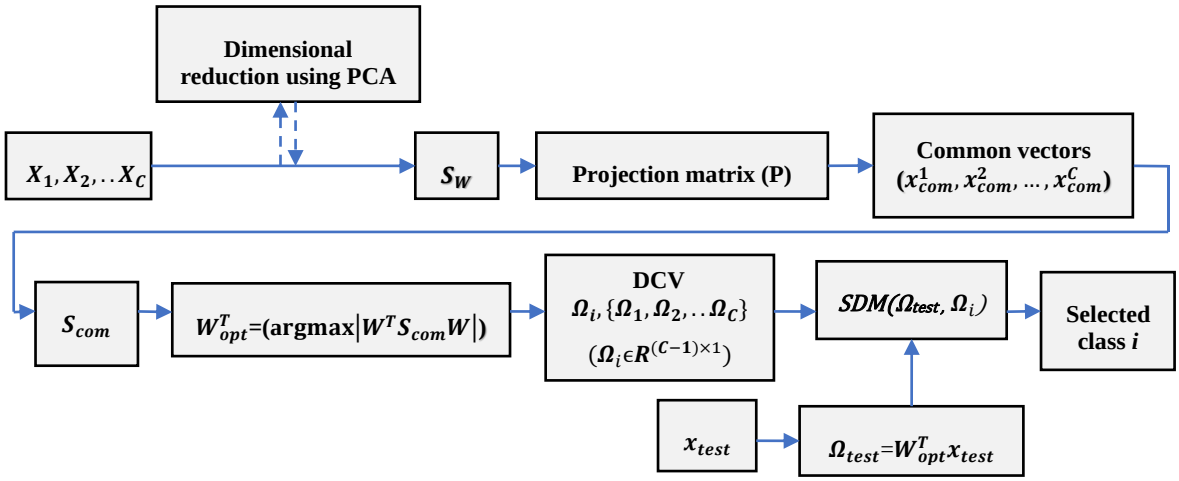

Figure 4. Block diagram of the proposed DCVA algorithm

The vectors **X** in Figure 4 are the feature vectors obtained using the MFCC or PLP coefficients. Tests were performed both with and without PCA for DCVA. When PCA is used for both DCVA and FLDA, a better recognition rate can be obtained by eliminating the noise components contained in the feature vectors. The within-class scatter matrix $\boldsymbol{S_w}$ is found as follows:

$$\boldsymbol{S_w} = \sum_{i=1}^{C} \sum_{j=1}^{K} \left((\mathbf{x}_j^i - \boldsymbol{\mu_i})(\boldsymbol{x_j^i} - \boldsymbol{\mu_i})^T \right) \tag{4}$$

where μ_i denotes the mean vector of the *i*th class. To project the samples in the training set, we can use the eigenvectors consisting of the indifference subspace denoted by **U** (Gulmezoglu et al., 1999). Then, the projection matrix **P** is determined by,

$$\mathbf{P} = \mathbf{U}\mathbf{U}^{\mathrm{T}} \tag{5}$$

The common vectors are obtained for all classes as follows,

$$\boldsymbol{x_{com}^i} = P\boldsymbol{x_j^i}, \quad r=1,2,\dots,K, i=1,2,\dots,C \tag{6}$$

The scatter matrix obtained from common vectors can be found using the following equation:

$$\mathbf{S_{com}} = \sum_{i=1}^{C} (\boldsymbol{x_{com}^i} - \boldsymbol{\mu_{com}}) (\boldsymbol{x_{com}^i} - \boldsymbol{\mu_{com}})^T \tag{7}$$

where $\boldsymbol{\mu_{com}}$ indicates the mean vector of the common vectors. In this case, the eigenvectors corresponding to the nonzero eigenvalues of the $\mathbf{S_{com}}$ matrix give the optimal projection vectors for the DCVA.

$$J(\mathbf{W_{opt}}) = argmax|\mathbf{W}^T \mathbf{S_{com}} \mathbf{W}| \tag{8}$$

The feature vectors can be written by using the optimal projection vectors.

$$\boldsymbol{\Omega_i} = \left[< \mathbf{x_r^i}, \mathbf{w_1} > \cdots < \mathbf{x_r^i}, \mathbf{w_{C-1}} >\right] \tag{9}$$

These vectors ($\boldsymbol{\Omega_i}$) are called discriminative common vectors, whose dimensions are at most *C-1*. In the test phase, to classify the face images in the test set, the feature vectors of these images are found by

$$\boldsymbol{\Omega_{test}} = \mathbf{W_{opt}^T} \boldsymbol{x_{test}} \tag{10}$$

where $\mathbf{W_{opt}^T}=[\mathbf{w_1}\ \mathbf{w_2}\ldots\ \mathbf{w_{C\text{-}1}}]^T$ and $\boldsymbol{\Omega_{test}} \in \mathbf{R^{(C-1)\times 1}}$. The above operations were performed for the insufficient data case (M<n). However, in the case of sufficient data (M>n), difference and indifference subspaces can be determined by estimation (Gulmezoglu et al., 1999). Then, the distance values between $\boldsymbol{\Omega_{test}}$ and the feature vector $\boldsymbol{(\Omega_i)}$ of the *i*th class are found for the distance measures.

$$\boldsymbol{D_{a,i}} = Distance(\boldsymbol{\Omega_i}, \boldsymbol{\Omega_{test}})_a,\ i=1,2,\ldots,C \tag{11}$$

where a is the selected distance measures (Euclidean, Correlation, Cityblock, Spearman) and $D_{a,i}$ shows the distance value for the *i*th class and *a*th distance criterion. Finally, the vector $\boldsymbol{\Omega_{test}}$ is assigned to the class that gives the best suitable distance value.

2.6. The Proposed FLDA-KNN Algorithm

The FLDA algorithm maximizing separation between classes in the training process is one of the popular classifiers derived from Linear Discriminant Analysis (LDA) (Yang and Chen, 2014). Figure 5 illustrates the block diagram of the proposed FLDA-KNN algorithm, including the PCA algorithm.

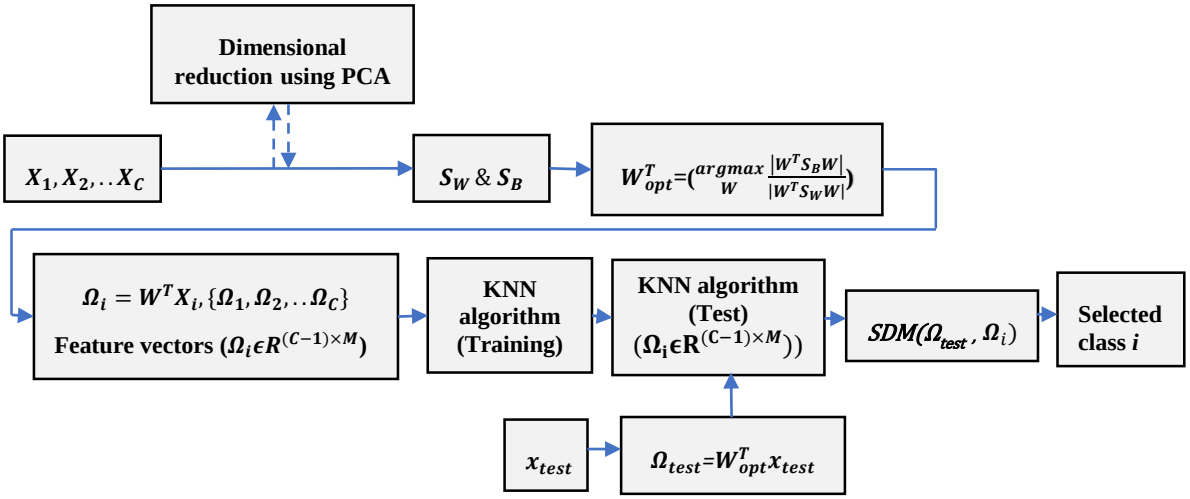

Figure 5. Block diagram of the proposed FLDA-KNN algorithm

As shown in Fig. 5, $\mathbf{S_w}$ and $\mathbf{S_B}$ scattering matrices are found for the training set's feature vectors. The between-class scatter matrix $\mathbf{S_B}$ is calculated by

$$\mathbf{S_B} = \sum_{i=1}^{C} N(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \tag{12}$$

where N is the number of samples in a class, $\boldsymbol{\mu_i}$ is the mean of the *i*th class, and $\boldsymbol{\mu}$ represents the mean of all classes. Then, the optimal set of basis vectors ($\mathbf{W_{opt}}$) is determined using these matrices (Belhumeur et al., 1997).

$$\mathbf{W_{opt}} = \underset{\mathbf{W}}{\mathbf{argmax}} \frac{|W^T S_B W|}{|W^T S_W W|}, \tag{13}$$

In Eq. (11) specified as below, *C-1* eigenvectors corresponding to the largest eigenvalues of the formed matrix as a result of $\mathbf{S_W^{-1}S_B}$ multiplication gives the optimal basis vector ($\mathbf{W_{opt}}$).

$$\boldsymbol{S_B w_i} = \boldsymbol{\lambda_i S_W w_i}, \quad i=1, 2, \ldots, m \tag{14}$$

Where m is equal to *C-1*. By finding this basis vector, all the feature vectors in the training set are projected onto the optimum space. In other words, the feature vectors with *C-1×M* dimensional are obtained for all samples in a class ($\boldsymbol{\Omega_i \epsilon R^{(C-1)\times M}}$). Then, using $\mathbf{W_{opt}}$, feature vectors are found for each class as follows,

$$\boldsymbol{\Omega}_i = \mathbf{W}_{\mathbf{opt}}^T \mathbf{X_i}, \quad i = 1,2, \ldots, C \tag{15}$$

where $\boldsymbol{\Omega}_i \in \mathbf{R}^{(\mathbf{C-1})\times \mathbf{K}}$ and, *K* is the number of samples in the *i*th class. The training process is carried out using the KNN algorithm for the feature vectors ($\boldsymbol{\Omega}_i$). For classification, the test signal is first projected using the $\mathbf{W_{opt}}$ and, then $\boldsymbol{\Omega}_{test}$ ($\boldsymbol{\Omega}_{test} \in \mathbf{R}^{(\mathbf{C-1})\times \mathbf{1}}$) is found by,

$$\boldsymbol{\Omega_{test}} = \mathbf{W}_{\mathbf{opt}}^T \boldsymbol{x_{test}} \tag{16}$$

In the test phase, the projected test signal ($\boldsymbol{\Omega_{test}}$) is found by using $\mathbf{W_{opt\ and}}$ assigned using the KNN algorithm to the most appropriate class. For example, Eq. (11) was given in the DCVA and is used for this assignment.

3. EXPERIMENTAL STUDIES

3.1. Speech Database

Speech commands dataset version-2 was used as the database, and 150 speech signals were used for each class. This speech database includes speech signals created by adding artificial mathematical as white gaussian noise and background noises in the study. Each speech signal is a single-channel and 16-bit-PCM signal with a sampling frequency of 16 kHz. These speech signals include numbers from zero to nine: 'zero', 'one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine' and contains 8 words such as 'follow', 'forward', 'stop', 'happy', 'house', 'learn', 'left', 'right'. The speech signals for each class were manually arranged in equal length. The length of each signal is 8800 samples and corresponds to 0.55 seconds. The MFCC and PLP coefficients were found, whose dimensions are 13 and 39 for each frame. Therefore, 689-dimensional (13×53) and 2067-dimensional (39×53) feature vectors (or feature matrices) were obtained for each speech signal. Then, these feature vectors (or feature matrices) were trained and tested separately. Figure 6 below shows nine speech signals containing various noises belonging to the word "five".

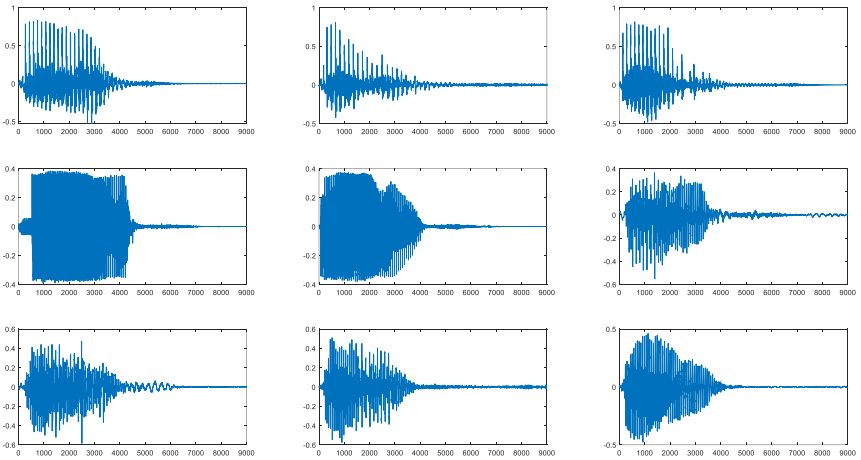

Figure 6. Speech signals containing noise belonging to the word 'five'

3.2. The Proposed Distance Measures

The following four distance measures were preferred in our experimental study. One is the Cityblock distance measure which finds the sum of the absolute values of the differences between two vectors (Abu Alfeilat et al., 2019). The Cityblock distance, which is always greater than or equal to zero, is given by Eq. 20.

$$\sum_{i=1}^{k} |a_i - b_i| \tag{20}$$

where k is the length of vectors a and b, another distance criterion, Euclidean, is given as Eq. 21 for vectors a and b.

$$\left(\sum_{i=1}^{k} (a_i - b_i)^2 \right)^{1/2} \tag{21}$$

The Spearman correlation coefficient is the Pearson correlation coefficient between the rank variables (Myers and Well, 2003). The Spearman correlation coefficient can be computed as follows,

$$r = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \tag{22}$$

Where $d_i = R(X_i) - R(Y_i)$ is the difference between the two ranks of each observation, n is the number of observations. The correlation distance (Székely et al., 2007) of two random variables is obtained by dividing their distance covariance by the product of their distance standard deviations. The correlation distance is,

$$d = 1 - \frac{(a - \bar{a})(b - \bar{b})^T}{\sqrt{(a - \bar{a})(a - \bar{a})^T} \sqrt{(b - \bar{b})(b - \bar{b})^T}} \tag{23}$$

Where $\bar{a} = \frac{1}{n} \sum_{j=1}^{n} a_j$ and $\bar{b} = \frac{1}{n} \sum_{j=1}^{n} b_j$ 'dir.

3.3. Experimental Results

All the studies were carried out in the MATLAB environment. In the study, 150 speech signals were used for each class. One feature matrix was found for each speech signal, that is, the number of the feature matrices is 150 for a class. Next, the feature matrices were divided into three equal parts with 50 dimensions; one was used for testing, and the other was used for training. Thus, 3-fold cross-validation was used for the classifiers. This process was repeated three times, and the test process was completed, and then the average recognition rates were found. The feature matrices for SVM, KNN, DCVA, and FLDA were converted into 689 and 2067-dimensional feature vectors used for the classification. In addition, for the proposed RNN-LSTM and CNN, training and testing processes were carried out using the feature matrices whose dimensions are 13×53 and 39×53. The study performed testing for the KNN, DCVA, and FLDA using Euclidean, Cityblock, Correlation, and Spearman distance measures. Besides, the numbers of the nearest neighbor for the KNN and FLDA-KNN classifiers were chosen as 1, 3, and 5 (K=1, K=3, and K=5). The study used RBF, polynomial, and linear kernels, the three most used kernels in the literature, for SVM. Adam (adaptive moment estimation) optimizer was used for training the network with RNN-LSTM. The number of epochs used in the training phase was 100, the number of iterations was 6600, iterations per epoch was 66, and the learning rate was 0.001. Besides, this algorithm has the LSTM with 100 hidden units, a Fully Connected layer, SoftMax and Classification Output. The proposed RNN-LSTM model's layers are listed in Table 1 below.

Table 1. The layers of the proposed RNN model

Layer level	Layers
1	Sequence Input (Sequence input with 13 or 39 dimensions)
2	LSTM (LSTM with 100 hidden units)
3	Fully Connected (C fully connected layer, C is the number of classes)
4	Softmax
5	Classification Output

Adam (adaptive moment estimation) optimizer was used to train the CNN network. The number of epochs used in

the training phase was chosen as 100, iterations per epoch are 14, and the learning rate is 0.0003. For the proposed CNN and RNN-LSTM, confusion matrices obtained using 50 test data and MFCC13 are given in Fig. 7 (a) and (b) below. In addition, the numbers corresponding to the classes in the confusion matrix are given below.

"1:eight","2:five","3:follow","4:forward","5:four","6:happy","7:house","8:learn","9:left","10:nine","11:one","12:right","13:seven","14:six","15:stop","16:three","17:two","18:zero".

True Class \ Predicted Class	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	50																	
2		48								1		1						
3		6	36	4									4					
4		1	4	36	5			1					2				1	
5			3	2	44									1				
6		1				47	1						1					
7							50											
8			1					42		2				1				4
9		2			2			1	40	1		2						
10	1						1			44	3	1						
11								1		3	46							
12	2	1						1	3	1	4	38						
13						1		1					47				1	
14						1								48		1		
15		1			1				1		1		2		44			
16	1															40	9	
17	1				1	1							2			4	40	1
18					1		1									2		46

(a)

True Class \ Predicted Class	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	44								3		3							
2	1	46		2							1							
3			43	5						2								
4			2	41	5						1		1					
5			1	6	42					1								
6						50												
7							50											
8		1						42		5	2							
9								5	45									
10		3								45		2						
11					2						48							
12	1										1	48						
13						1							47				2	
14														50				
15			1			1									48			
16	1															48	1	
17													1				48	1
18																	2	48

(b)

Figure 7. The recognition values for the CNN (%87.33) (a) and RNN-LSTM (%92.56) (b)

Recognition results obtained using the DCVA and FLDA-KNN are found for sufficient data cases. For sufficient data, the ratio of the sum of the k smallest eigenvalues to the sum of all eigenvalues gives the amount of energy used. The study observed high recognition rates when energy of around 30% was used for the DCVA. The average recognition ratios of the proposed DCVA and FLDA-KNN are given in Table 2 for 13 MFCC coefficients ($MFCC_{13}$).

Table 2. The results of the proposed DCVA and FLDA-KNN algorithms for $MFCC_{13}$

Distance Measures	DCVA	FLDA-KNN (K=1)	FLDA-KNN (K=3)	FLDA-KNN (K=5)
Euc	74.23	71.03	70.51	71.25
Corr	73.56	67.41	67.22	68.07
City	71.45	67.83	68.78	69.14
Sp	63.78	61.42	60.22	61.97

Table 3. The results of the proposed DCVA and FLDA-KNN algorithms for PLP_{13}

Distance Measures	DCVA	FLDA-KNN (K=1)	FLDA-KNN (K=3)	FLDA-KNN (K=5)
Euc	76.40	69.30	69.66	71.78
Corr	73.22	67.14	67.55	69.45
City	73.93	67.83	68.78	70.30
Sp	55.18	61.42	62.22	64.85

When the results in Tables 4 and 5 below are examined, it is seen that $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ algorithms give higher recognition rates than DCVA and FLDA-KNN algorithms in Table 2 and Table 3. In Tables 2 and 3, the best DCVA and FLDA-KNN (K=5) results for the Euclidean distance measure were 76.40% and 71.78%, respectively. In Tables 4 and 5, the best DCVA and FLDA-KNN (K=5) results were found at 79.00% and 84.90%, respectively. While 79.00% recognition rate was found using $MFCC_{13}$ and Euclidean distance measure, 84.90% was found using PLP_{13} and Correlation distance measure.

Table 4. The results of the proposed $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ algorithms for $MFCC_{13}$

Distance Measures	$(DCVA)_{PCA}$	$(FLDA\text{-}KNN)_{PCA}$ (K=1)	$(FLDA\text{-}KNN)_{PCA}$ (K=3)	$(FLDA\text{-}KNN)_{PCA}$ (K=5)
Euc	76.44	80.11	81.34	82.44
Corr	73.54	79.11	79.55	80.41
City	72.23	79.70	80.89	81.20
Sp	54.36	70.44	71.55	72.44

Table 5. The results of the proposed $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ algorithms for PLP_{13}

Distance Measures	$(DCVA)_{PCA}$	$(FLDA\text{-}KNN)_{PCA}$ (K=1)	$(FLDA\text{-}KNN)_{PCA}$ (K=3)	$(FLDA\text{-}KNN)_{PCA}$ (K=5)
Euc	79.00	82.67	82.85	84.22
Corr	75.56	81.44	81.56	84.90
City	76.88	81.00	81.44	82.55
Sp	56.45	72.88	73.77	75.44

Recognition rates for KNN are given in Tables 6 and 7. Tables 6 and 7 show that the highest recognition values obtained using the correlation distance measure are 78.91% and 86.51%. These recognition values were obtained using $MFCC_{13}$ and $MFCC_{39}$. In addition, the recognition rates found for Spearman and Correlation distance measures are higher than those of Euclidean distance.

Table 6. The results of the proposed KNN algorithms for $MFCC_{13}$ and PLP_{13}

	$MFCC_{13}$			PLP_{13}		
Distance Measures	**K=1**	**K=3**	**K=5**	**K=1**	**K=3**	**K=5**
Euc	67.88	75.62	76.37	67.40	75.04	76.02
Corr	73.21	78.78	78.91	70.31	76.04	76.22
City	68.74	76.25	76.01	66.67	75.41	75.45
Sp	72.86	78.30	78.55	71.07	76.33	76.63

Table 7. The results of the proposed KNN algorithms for $MFCC_{39}$ and PLP_{39}

	$MFCC_{39}$			PLP_{39}		
DM	**K1**	**K3**	**K5**	**K1**	**K3**	**K5**
Euc	63.04	65.22	69.67	60.18	69.97	70.00
Corr	79.51	86.51	86.18	78.55	84.85	85.25
City	64.72	68.70	65.63	62.66	73.03	72.74
Sp	79.77	85.33	85.93	77.78	84.35	84.20

Table 8. The results of the proposed $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ algorithms for $MFCC_{39}$

Distance Measures	**DCVA**	**$(DCVA)_{PCA}$**	**$(FLDA\text{-}KNN)_{PCA}$ (K=1)**	**$(FLDA\text{-}KNN)_{PCA}$ (K=3)**	**$(FLDA\text{-}KNN)_{PCA}$ (K=5)**
Euc	73.44	73.67	78.55	79.67	80.94
Corr	70.11	71.11	78.00	78.67	79.78
City	70.22	70.11	77.89	78.22	79.66
Sp	55.00	55.67	61.44	62.21	64.34

In Table 9, the best $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ results were 68.44% and 77.29%, respectively.

Table 9. The results of the proposed DCVA, $(DCVA)_{PCA}$, and $(FLDA\text{-}KNN)_{PCA}$ algorithms for PLP_{39}

Distance Measures	**DCVA**	**$(DCVA)_{PCA}$**	**$(FLDA\text{-}KNN)_{PCA}$ (K=1)**	**$(FLDA\text{-}KNN)_{PCA}$ (K=3)**	**$(FLDA\text{-}KNN)_{PCA}$ (K=5)**
Euc	68.22	68.44	74.22	75.33	77.29
Corr	66.89	67.11	72.44	74.44	76.18
City	64.44	66.33	71.22	74.78	76.26
Sp	57.77	56.56	63.66	64.02	66.36

For the CNN and RNN-LSTM in Table 10, highest recognition rates are obtained 87.56% and 93.21%, respectively.

Table 10. ***The results of the proposed RNN-LSTM, CNN, and SVM algorithms***

Features	**CNN**	**RNN-LSTM**	**SVM-POL**	**SVM-RBF**	**SVM-LIN**
Plp_{13}	86.12	86.24	77.78	67.94	57.22
Plp_{39}	86.21	93.22	73.88	66.11	56.23
$MFCC_{13}$	87.56	88.45	76.56	66.63	57.11
$MFCC_{39}$	85.31	85.17	75.25	64.45	56.42

The highest recognition rates of all classifiers are given in Figure 8 below.

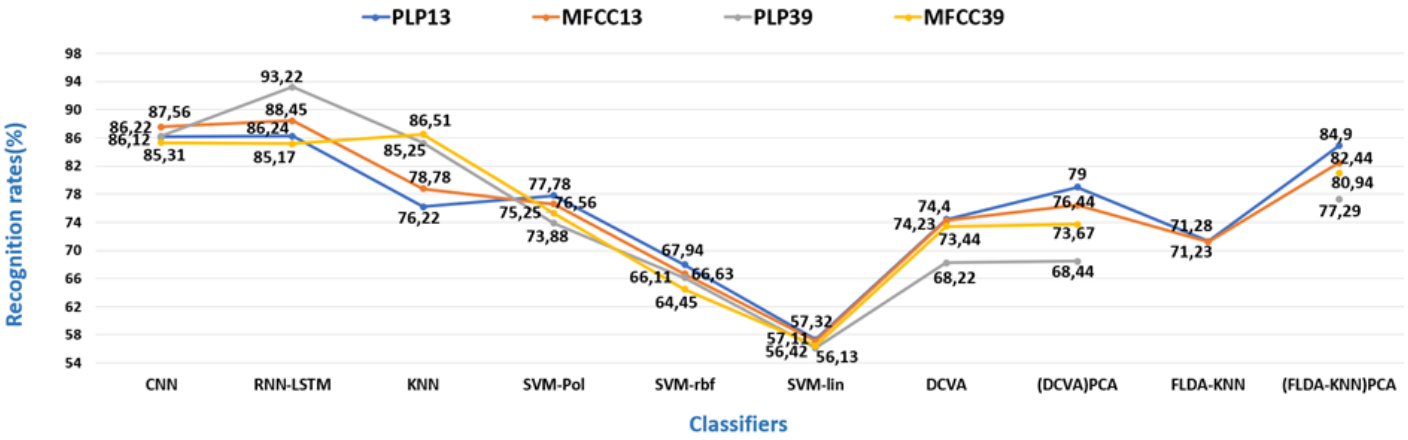

Fig 8. The highest recognition rates of the proposed classifiers

Speech recognition experiments on the testing set are depicted in Fig. 8. As shown in Fig. 8, the CNN and RNN-LSTM achieved over 85% accuracy. On the other hand, the lowest recognition rate belongs to SVM-linear (57.11%).

4. DISCUSSION AND CONCLUSIONS

Comprehensive tests are carried out for the speech signals with noise in the study. Subspace classifiers and machine learning classification algorithms were used for the test process. While only one distance measure was used in many classical studies, four different distance measures were used in this study to make a more in-depth analysis. The study's MFCC and PLP feature vectors have 13 or 39 sizes per frame.

A hybrid FLDA-KNN algorithm was also performed using the FLDA and KNN classifiers. The DCVA was deeply tested for the first time in isolated word recognition. Besides, the hybrid new $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ algorithms were performed using PCA. The highest recognition rates of the DCVA and FLDA-KNN were obtained 74.4% and 71.28%, respectively. On the other hand, $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ classifiers were better results than DCVA and FLDA-KNN classifiers. The highest recognition rates for $(DCVA)_{PCA}$ and $(FLDA\text{-}KNN)_{PCA}$ were 79% and 84.9%, respectively. Also, the correlation distance measure gave the best results for FLDA-KNN.

CNN and RNN-LSTM, machine learning algorithms, gave better results than subspace methods. The highest recognition rates for RNN-LSTM and CNN are 93.21% and 87.56%, respectively. For RNN-LSTM, the best recognition rate was found using 39-dimensional PLP feature vectors. KNN and SVM, other machine learning algorithms, gave 86.51% and 77.78%, respectively. Especially for KNN, it was observed that the correlation distance criterion gave better recognition results than Euclidean. While CNN and RNN-LSTM gave high recognition rates for all feature vectors, subspace methods gave better results, especially for low-dimensional $MFCC_{13}$ and PLP_{13} feature vectors. These results showed that the RNN-LSTM and CNN gave more satisfactory recognition rates than other classifiers. When the experimental results were examined, it was seen that the distance criterion to be used significantly affected the recognition rates.

Competing interests

The authors declare that they have no competing interests.

Data Availability Statements: The datasets and codes used in the current study are available in the github repository,

https://github.com/solaris3344/Hybrid-subspace-and-ML

REFERENCES

Abdel-Hamid, O., & Jiang, H. (2013, May). Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code. In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (pp. 7942-7946). IEEE.

Abu Alfeilat, H. A., Hassanat, A. B., Lasassmeh, O., Tarawneh, A. S., Alhasanat, M. B., Eyal Salman, H. S., & Prasath, V. S. (2019). Effects of distance measure choice on k-nearest neighbor classifier performance: a review. Big data, 7(4), 221-248.

Akyazi, Ö., Şahin, E., Özsoy, T., & Algül, M. (2019). A Solar Panel Cleaning Robot Design and Application. Avrupa Bilim ve Teknoloji Dergisi, 343-348.

Anggraeni, D., Sanjaya, W. S. M., Nurasyidiek, M. Y. S., & Munawwaroh, M. (2018). The implementation of speech recognition using mel-frequency cepstrum coefficients (MFCC) and support vector machine (SVM) method based on python to control robot arm. In IOP Conference Series: Materials Science and Engineering (Vol. 288, No. 1, p. 012042). IOP Publishing.

Beigi, H. (2011). Speaker recognition. In Fundamentals of Speaker Recognition (pp. 543-559). Springer, Boston, MA.

Belhumeur P. N., Hespanha J. P., Kriegman D. J., "Eigenfaces vs fisherfaces: Recognition using class specific linear projection," IEEE Trans. on PAMI, Vol. 19, No:7, pp. 711-720, 1997.

Bharali, S. S., & Kalita, S. K. (2015). A comparative study of different features for isolated spoken word recognition using HMM with reference to Assamese language. International Journal of Speech Technology, 18(4), 673-684.

Cevikalp Hakan et al., "Discriminative common vectors for face recognition", Pattern Analysis and Machine Intelligence IEEE Transactions on 27, vol. 1, pp. 4-13, 2005.

Dokuz, Y., & Tüfekci, Z. (2020). A Review on Deep Learning Architectures for Speech Recognition. Avrupa Bilim ve Teknoloji Dergisi, 169-176.

Filho, G. L., & Moir, T. J. (2010). From science fiction to science fact: a smart-house interface using speech technology and a photo-realistic avatar. International journal of computer applications in technology, 39(1-3), 32-39.

Furui, S., Kikuchi, T., Shinnaka, Y., & Hori, C. (2004). Speechto-text and speech-to-speech summarization of spontaneous speech. IEEE Transactions on Speech and Audio Processing, 12(4), 401-408.

Gaikwad, S. K., Gawali, B. W., & Yannawar, P. (2010). A review on speech recognition technique. International Journal of Computer Applications, 10(3), 16-24.

Garcia, C. I., Grasso, F., Luchetta, A., Piccirilli, M. C., Paolucci, L., & Talluri, G. (2020). A comparison of power quality disturbance detection and classification methods using CNN, LSTM and CNN-LSTM. Applied Sciences, 10(19), 6755.

Gulati, A., Qin, J., Chiu, C. C., Parmar, N., Zhang, Y., Yu, J., ... & Pang, R. (2020). Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint arXiv:2005.08100.

Gulmezoglu, M. B., Dzhafarov, V., Keskin, M., & Barkana, A. (1999). A novel approach to isolated word recognition. IEEE Transactions on Speech and Audio Processing, 7(6), 620-628.

Gulmezoglu, M. B., Edizkan, R., Ergin, S., & Barkana, A. (2005, May). Improvements on isolated word recognition using FLDA. In Proceedings of the IEEE 13th Signal Processing and Communications Applications Conference, 2005. (pp. 703-706). IEEE.

Gunal, S., & Edizkan, R. (2008). Subspace based feature selection for pattern recognition. Information Sciences, 178(19), 3716-3726.

Haque, M. A., Verma, A., Alex, J. S. R., & Venkatesan, N. (2020). Experimental evaluation of CNN architecture for speech recognition. In First international conference on sustainable technologies for computational intelligence (pp. 507-514). Springer, Singapore.

Imtiaz, M. A., & Raja, G. (2016, November). Isolated word automatic speech recognition (ASR) system using MFCC,

DTW & KNN. In 2016 asia pacific conference on multimedia and broadcasting (APMediaCast) (pp. 106-110). IEEE.

Keser, S., & Edizkan, R. (2009, April). Phonem-based isolated Turkish word recognition with subspace classifier. In 2009 IEEE 17th Signal Processing and Communications Applications Conference (pp. 93-96). IEEE.

Kolossa, D., Zeiler, S., Saeidi, R., & Astudillo, R. F. (2013). Noise-adaptive LDA: A new approach for speech recognition under observation uncertainty. IEEE Signal Processing Letters, 20(11), 1018-1021.

Lalitha, S., Mudupu, A., Nandyala, B. V., & Munagala, R. (2015, December). Speech emotion recognition using DWT. In 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (pp. 1-4). IEEE.

Miao, F., Zhang, P., Jin, L., & Wu, H. (2018, August). Chinese news text classification based on machine learning algorithm. In 2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC) (Vol. 2, pp. 48-51). IEEE.

Mohan, B. J. (2014, January). Speech recognition using MFCC and DTW. In 2014 International Conference on Advances in Electrical Engineering (ICAEE) (pp. 1-4). IEEE.

Muhammad, H. Z., Nasrun, M., Setianingsih, C., & Murti, M. A. (2018, May). Speech recognition for English to Indonesian translator using hidden Markov model. In 2018 International Conference on Signals and Systems (ICSigSys) (pp. 255- 260). IEEE.

Myers, Jerome L.; Well, Arnold D. (2003). Research Design and Statistical Analysis (2nd ed.). Lawrence Erlbaum. pp. 508. ISBN 978-0-8058-4037-7.

Najkar, N., Razzazi, F., & Sameti, H. (2010). A novel approach to HMM-based speech recognition systems using particle swarm optimization. Mathematical and Computer Modelling, 52(11-12), 1910-1920.

Palaz, D., & Collobert, R. (2015). Analysis of cnn-based speech recognition system using raw speech as input (No. REP_WORK). Idiap.

Palaz, D., Magimai-Doss, M., & Collobert, R. (2019). End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition. Speech Communication, 108, 15-32.

Passricha, V., & Aggarwal, R. K. (2020). A hybrid of deep CNN and bidirectional LSTM for automatic speech recognition. Journal of Intelligent Systems, 29(1), 1261-1274.

Permanasari, Y., Harahap, E. H., & Ali, E. P. (2019, November). Speech recognition using dynamic time warping (DTW). In Journal of Physics: Conference Series (Vol. 1366, No. 1, p. 012091). IOP Publishing.

Seltzer, M. L., Yu, D., & Wang, Y. (2013, May). An investigation of deep neural networks for noise robust speech recognition. In 2013 IEEE international conference on acoustics, speech and signal processing (pp. 7398-7402). IEEE.

Sivaram, G. S., Nemala, S. K., Mesgarani, N., & Hermansky, H. (2010). Data-driven and feedback based spectro-temporal features for speech recognition. IEEE Signal Processing Letters, 17(11), 957-960.

Song, Y., Huang, J., Zhou, D., Zha, H., & Giles, C. L. (2007, September). Iknn: Informative k-nearest neighbor pattern classification. In European Conference on Principles of Data Mining and Knowledge Discovery (pp. 248-264). Springer, Berlin, Heidelberg.

Song, K. T., Han, M. J., & Wang, S. C. (2014). Speech signal-based emotion recognition and its application to entertainment robots. Journal of the Chinese Institute of Engineers, 37(1), 14-25.

Soucy, P., & Mineau, G. W. (2001, November). A simple KNN algorithm for text categorization. In Proceedings 2001 IEEE international conference on data mining (pp. 647-648). IEEE.

Soujanya, M., & Kumar, S. (2010, August). Personalized IVR system in contact center. In 2010 International Conference on Electronics and Information Engineering (Vol. 1, pp. V1- 453). IEEE.

Speech commands dataset version 2 (2018). [Online]. Available:http://download.tensorflow.org/data/speech_commands_v0.02.tar.gz

Srisuwan, N., Phukpattaranont, P., & Limsakul, C. (2018). Comparison of feature evaluation criteria for speech

recognition based on electromyography. Medical & biological engineering & computing, 56(6), 1041-1051.

Sumit, S. H., Al Muntasir, T., Zaman, M. A., Nandi, R. N., & Sourov, T. (2018, September). Noise robust end-to-end speech recognition for bangla language. In 2018 international conference on bangla speech and language processing (ICBSLP) (pp. 1-5). IEEE.

Székely, G. J., Rizzo, M. L., & Bakirov, N. K. (2007). Measuring and testing dependence by correlation of distances. The annals of statistics, 35(6), 2769-2794.

Tan, K., & Wang, D. (2018, September). A Convolutional Recurrent Neural Network for Real-Time Speech Enhancement. In Interspeech (Vol. 2018, pp. 3229-3233).

Tan, T., Qian, Y., Hu, H., Zhou, Y., Ding, W., & Yu, K. (2018). Adaptive very deep convolutional residual network for noise robust speech recognition. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 26(8), 1393-1405.

Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T., & Kitamura, T. (2000, June). Speech parameter generation algorithms for HMM-based speech synthesis. In 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100) (Vol. 3, pp. 1315-1318). IEEE.

Wahyuni, E. S. (2017, November). Arabic speech recognition using MFCC feature extraction and ANN classification. In 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE) (pp. 22-25). IEEE.

Yang, L., & Chen, S. (2014). Linear discriminant analysis with worst between-class separation and average within-class compactness. Frontiers of Computer Science, 8(5), 785-792.

Yavuz, H. S., Çevikalp, H., & Barkana, A. (2006). Twodimensional CLAFIC methods for image recognition. In 2006 IEEE 14th Signal Processing and Communications

Zhang, S. X., & Gales, M. J. (2012). Structured SVMs for automatic speech recognition. IEEE Transactions on Audio, Speech, and Language Processing, 21(3), 544-555.