IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 4, 2021, accepted May 20, 2021, date of publication May 27, 2021, date of current version June 7, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3084200

Use of Machine Learning for Deception Detection From Spectral and Cepstral Features of Speech Signals

SINEAD V. FERNANDES, (Graduate Student Member, IEEE), AND MUHAMMAD S. ULLAH, (Senior Member, IEEE)
Department of Electrical and Computer Engineering, Florida Polytechnic University, Lakeland, FL 33805, USA
Corresponding author: Muhammad S. Ullah (mullah@floridapoly.edu)

ABSTRACT In this research, four unique nonlinear speech features are extracted and analyzed to study the dissimilarity pattern between when the speaker is being deceitful and truthful based on how human speech is perceived. The speaker was under stress in a police interrogation where two ground truth and two deceitful responses were recorded during three different times of the day. Using the audio recordings from all three sessions, the cepstral features and spectral energy features are extracted. Cepstral features are the Mel frequency cepstrum coefficient, from where the delta cepstrum and the time-difference cepstrum features are developed. On the other hand, the spectral energy features are the energy of Bark band energy from where the delta energy and the time-difference energy features are developed. The Levenberg-Marquardt classification method and the long short-term memory classification method are then applied to evaluate the accuracy of detecting deception based on the nine unique training and testing combinations of the three different sessions and their extracted cepstrum and spectral energy features. In addition, the principal component analysis is applied to reduce the dimensionality from the extracted features for further improvement. The projected principal components of the four types of features showed improved accuracy in order to distinguish between truthful and deceptive speech pattern. After incorporating with principal component analysis, the long short-term memory classification method with time-difference spectral energy feature shows the highest recognition rate compared to Levenberg-Marquardt algorithm with other cepstral and spectral features.

INDEX TERMS Cepstral features, deception detection, machine learning, principal component analysis, spectral features, speech analysis.

I. INTRODUCTION

Deception detection is considerable practical interest in the field of law enforcement and other government agencies to identify the potential deception at the border crossing and in military scenarios for national security applications. It is also used to evaluate reports from informants at embassies and consulates throughout the world [6], [36]. Deception is intentionally causing an individual to accept falsehood as one that is true. Psychologically speaking someone is being deceptive when subconscious or conscious movements occur including the shortened length of speech, a flushed face, changes in the voice frequency, avoidant eye contact, changes in the diameter of the eye pupil, and a more rigid body [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

Deception is incorporated in everyday interactions, yet it is challenging for untrained and trained professionals [2], [3] to accurately detect it without the use of intrusive measures. Those being commonly used polygraph technology that measured the changes in respiratory rate, electrodermal activity (sweatiness of fingertips), blood pressure, and heart rate [3] using blood pressure cuffs, rubber tubes, and metal plates attached to the fingers. It is also well known that human speech has emotion and nonlinguistic information encoded in it. People who are aiming to be deceptive typically present minuscule changes in sound pressure, vocal organs, tone, speed of speech, and increased pause time as compared to that of a person being truthful [15]. As such, studying these changes can lead to detecting deception with accuracy. In recent years, researchers have been studying the multitude of ways in which humans present their deceptive tricks.

The research includes speech signal analysis [2], [4], [6], [12]–[15], thermal facial analysis [7], text analysis with BERT [8], and visual cue analysis [9]–[11] which have proven to be viable options to detect deception. The principal difference between the above technologies is that the polygraph measures some human responses like respiratory rate, electro-dermal activity, heart rate, and others through direct contact, which can lead to numerous challenges and complexities in terms of implementation. As a result, how well humans or machines may ultimately perform at the task of detecting deceptive speech continues to remain a challenging question.

Speech-based deception detection devices will provide a less invasive option, are inexpensive to produce, and can operate effortlessly. It could also be used for non-present subjects which has considerable advantages and has great potential to be beneficial in real-world applications. Therefore, in this research, we investigate an unintrusive alternative to detect the deception using speech signal-based features on a subject in a high-stress (i. e. police interrogation) environment. This research explores the cepstral features in terms of proposed delta cepstrum and time-difference cepstrum, as well as spectral energy features in terms of proposed delta energy and time-difference energy to analyze and distinguish between deceptive and truthful speech. The Levenberg-Marquardt algorithm and the Long Short-Term Memory (LSTM) neural network models are classification methods that are used to test the recognition rate using all the four extracted features.

Although these techniques and classification methods are well known separately in the domain of artificial intelligence and signal processing, however, investigating different spectral and cepstral features (i. e. Mel frequency cepstrum coefficients (MFCCs) [6], [36], [37], [40], spectral roll of point [40], spectral flux [19], [38], [40], spectral centroid [40], spectral compactness, FFT [16], spectral variability, linear prediction cepstrum coefficients [44], logarithmic of energy, fundamental frequency and zero-crossing rate [40], [43]), develop new features (delta cepstrum, time difference cepstrum, delta energy, time difference energy) in conjunction with principal component analysis (PCA), and find an innovative solution at top-level design approach are key novelty of this research work. This research is also aimed to optimize the features and neural network classifiers to achieve a better recognition rate. In the early stage, for example, the PCA was used in a diabetes disease prediction and achieved with an increased rate of accuracy at 82.1% [5] which inspires us to apply it in this research. By applying PCA to the cepstrum features and the spectral energy features, a higher recognition rate is achieved to detect deception which is the main objective of this paper. Fig. 1 shows the overall proposed deception detection workflow from top-level design approach.

The remainder of the paper is organized as follows. The challenges and issues on deception detection based on previous research work are discussed in Section II. Section III shows the database that is used to experiment with this research. Section IV shows the cepstrum and spectral energy features extraction and analysis approach. The results of the projected principal component analysis of the previous data sets are presented in Section V. Section VI presents the feature matching techniques and their comparative results before and after applying the PCA. Finally, Section VII concludes the paper and discusses future research work.

II. PREVIOUS RESEARCH WORK

In recent years, numerous avenues have been studied to test the possibility of accurately detecting deception in humans using technological aid. Studies on deceptive behavior have analyses conducted on the facial, gestural, and biometric data as well as psychological evaluations on human perception [12]. They have all shown that these unique features can be used to recognize when a person is being deceptive to a certain extent. Challenges encountered include small/limited deceptive speech corpus, only intra-gender studies (single-gender), and low accuracy due to lack of training samples [2]. Ullah *et al*. extracted MFCC features to distinguish between deceptive speech and non-deceptive speech using the Levenberg-Marquardt Backpropagation algorithm [36]. The database used for that work is a collection of utterances from the audio recording of a male suspect under criminal investigation. Based on prosodic, lexical, and acoustic features and using the Columbia-SRI-Colorado (CSC) corpus, Graciarena *et al.* in SRI International proposed a Support Vector Machine (SVM) and Gaussian Mixture Model (GMM) combination system to detect deceptive and non-deceptive speech [12]. They also ran tests with the CSC corpus using a combination of acoustic GMM and prosodic/lexical SVM with similar accuracy results. Using a radial basis function neural network (RBFNN), SVM, and relevance vector machine (RVM), Zhou *et al.* tested eighteen (18) combined prosodic and non-linear dynamic (NLD) features as input to detect deception from speech signals [14]. Table 1 shows the different methods or algorithms used in previous research work to test unique speech features and how accurately they recognize deception.

III. EXPERIMENTAL DATABASE

For this research, the data being used was collected during a police investigation between a law enforcement officer and a guilty male criminal. The speaker was asked the same set of questions at three different times of the day. Two questions were designed to get non-deceptive responses and two questions were designed to get deceptive responses. The database used in this research is a set of speech utterances from audio recordings of a male criminal suspect during an interrogation by a law enforcement officer [6], [13]. During the polygraph testing, the criminal was asked a series of questions at three different times of day that resulted in accumulating twelve responses of 'No'. For analysis, six deceptive responses and six truthful responses were used. Within each of the three sessions, two sets of deceptive utterances labeled Q4 and Q5 as well as two sets of truthful utterances labeled Q7 and Q9

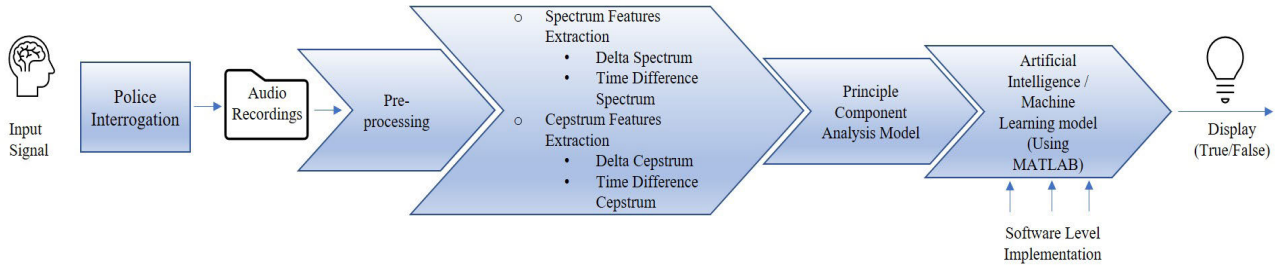

FIGURE 1. An overview of deception detection workflow from top level design approach.

TABLE 1. Feature and classification method comparison of deception detection techniques.

Ref	Feature	Type	Database	Classification Method	Recognition Rate
[18]	Bark Energy	Stressed Speech	Criminal Interrogation	Levenberg-Marquardt	83.33%
[6][18]	Significant Energy	Stressed Speech	Criminal Interrogation	Levenberg-Marquardt	66.67%
[12]	Prosodic-Lexical Combination	Speech	Columbia-SRI-Colorado (CSC) corpus	Support Vector Machine (SVM)	64.4%
[12]	Acoustic Spectral-based Mel cepstral features with energy	Speech	Columbia-SRI-Colorado (CSC) corpus	Gaussian Mixture Model (GMM)	62.1%
[39]	Acoustic/Prosodic, Lexical, and Speaker-Dependent Combination	Speech	Columbia-SRI-Colorado (CSC) corpus	*Ripper* Rule Induction Classifier	66.4%
[37]	MFCC	Speech	Columbia-SRI-Colorado (CSC) corpus	Support Vector Machine (SVM)	51.8%
[37]	MFCC	Speech	Columbia-SRI-Colorado (CSC) corpus	LSTM	54.6%
[37]	Energy	Speech	Columbia-SRI-Colorado (CSC) corpus	Support Vector Machine (SVM)	50.2%
[37]	Energy	Speech	Columbia-SRI-Colorado (CSC) corpus	LSTM	47%
[37]	MFCC, and Energy Combination	Speech	Columbia-SRI-Colorado (CSC) corpus	Ensemble	55.8%
[40]	LexRNN: Lexical features	Speech	Columbia-SRI-Colorado (CSC) corpus	LSTM	81.03%
[40]	HybridRNN: Speaker-dependent features and lexical features	Speech	Columbia-SRI-Colorado (CSC) corpus	LSTM	84.11%
[40]	AudioRNN: Speaker-dependent acoustic features	Speech	Columbia-SRI-Colorado (CSC) corpus	LSTM	62.59%
[36]	MFCC (Pre-PCA)	Stressed Speech	Criminal Interrogation	Levenberg-Marquardt	61.33%
[6]	MFCC (Pre-PCA)	Stressed Speech	Criminal Interrogation	BFGS Quasi-Newton	75%
[14]	The Prosodic features, Nonlinear Linear Dynamic features	Male Speech	University of Soochow corpus	RBFNN	42.13%
[14]	The Prosodic features, Nonlinear Linear Dynamic features	Male Speech	University of Soochow corpus	SVM	68.14%
[14]	The Prosodic features, Nonlinear Linear Dynamic features	Male Speech	University of Soochow corpus	RVM	70.37%
[2]	Short-time Energy (STE), Pitch, Format, and Duration features	Speech	Chinese Deception Detection corpus	Gradient Boosting Decision Tree (GBDT)	68.02%
[41]	Critical "hot spot" segments identified to tell if a speaker is truthful or lying	Speech	Columbia-SRI-Colorado (CSC) corpus	Bagging, AdaBoost, and J48 combination	68.6%
[42]	Lexical and Speech	Speech	Indonesian Deception corpus	Random Forest Decision Tree and Random Undersampling (RUS)	61.26%

were designated for reference and analysis. The speech utterances were sampled at a rate of 16000 samples per second. Fig. 2 shows the reference signals that are used to investigate the deception detection in this research.

All the voice signal data was then processed individually to extract unique sets of voice features for inspection. Feature extraction is important to test the data's importance and relevance in terms of being useful to detecting deception.

Once the relevant data is analyzed, it is passed through artificial neural network models to train and test the robustness of the features extracted.

IV. FEATURE DEVELOPMENT AND EXTRACTION

The purpose of feature engineering is to analyze and distinguish between deceptive and truthful utterances quantitatively and reliably from speech [2]. When someone is being dishonest, there is a variety of complicated combinations of specific emotions and speech characteristics that could be studied from these features. In this paper, we extracted two cepstral and two spectral energy features from the reference speech signals that is shown in Fig. 2. The two cepstrum features extracted are time-difference cepstrum, and delta cepstrum while the two spectral energy features extracted are time-difference energy, and delta energy.

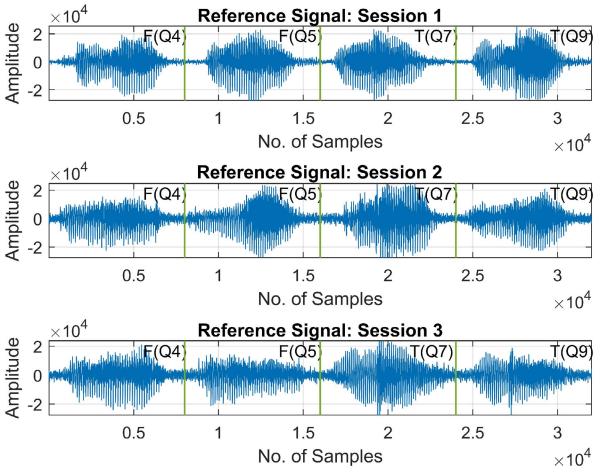

FIGURE 2. Reference speech signals.

A. CEPSTRUM FEATURE DEVELOPMENT AND EXTRACTION

Deception detection based on extracted cepstrum features was studied to understand how speech features can be used to detect human emotion and deception. Cepstral representation of an utterance provides a depiction of the local spectral properties of the signal [3], [6], [9]. Each speech utterance was divided into frames within a 20ms duration, with 320 samples, and a 50 percent overlap. Using a 1024-point DFT, the cepstrum features were calculated. Each frame was then divided into 24 critical bands ranging up to approximately 7500 Hz which illustrates in Table 2. To take out DC, the first band starts with the frequency resolution (DF). The power spectral density, global masking threshold, and quiet threshold were acquired after normalizing each frame to see which frame of speech was above the threshold of at least 3dB above the global masking threshold. The DFT points were doubled compared to the previous 512-DFT points for more points available to obtain more accurate results. Fig. 3 shows the initial pipeline of the cepstrum feature extraction process where the MFCC feature is extracted.

MFCCs are computed by summing up the weighted log energy magnitudes in a band around a center frequency as

TABLE 2. Cepstrum feature critical band filter bank.

Band Index	Critical Bandwidth(Hz)	Band Index	Critical Bandwidth(Hz)
1	DF-86	13	1723-1981
2	86-172	14	1981-2326
3	172-258	15	2326-2756
4	258-431	16	2756-3187
5	431-517	17	3187-3876
6	517-689	18	3876-4307
7	689-775	19	4307-4737
8	775-948	20	4737-5254
9	948-1120	21	5254-5957
10	1120-1292	22	5957-6460
11	1292-1464	23	6460-6977
12	1464-1723	24	6977-7585

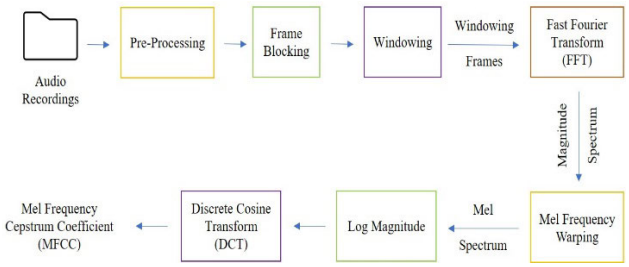

FIGURE 3. Cepstrum feature extraction pipeline.

shown in (1), where $n = 1, 2, \ldots K$ the number of cepstral coefficients, K is equal to the number of band index and S_k represents the Hamming window function used.

$$\mathrm{MFCC_n} = \sum_{k=1}^{K} \left(\log_{10} S_k\right) \cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{K}\right] \quad (1)$$

Once MFCC features are extracted, the delta cepstrum is calculated by taking the difference between successive MFCCs as shown in (2).

$$\text{delta cepstrum (n)} = \mathrm{MFCC_{n+1}} - \mathrm{MFCC_n} \quad (2)$$

The delta-cepstral features added to the static MFCC features strongly improves speech recognition accuracy [17]. For these reasons, some form of delta cepstral features is part of nearly all speech recognition systems.

The frame-to-frame difference cepstrum, to indicate the time-dependent variation, is obtained as a feature using the following expression that is shown in (3), where i indicate the number of frame index.

$$\text{difference cepstrum (m)} = \mathrm{MFCC_n\ (i+1)} - \mathrm{MFCC_n\ (i)} \quad (3)$$

For instance, using delta cepstrum features of critical band 11 and time-difference cepstrum features of critical band 18, Fig. 4 and Fig. 5 respectively show their feature domain representations of truthful and deceptive utterances of the word 'No'. The pattern of true 'NO' utterances is distinguishable from the pattern of deceptive 'NO' utterances.

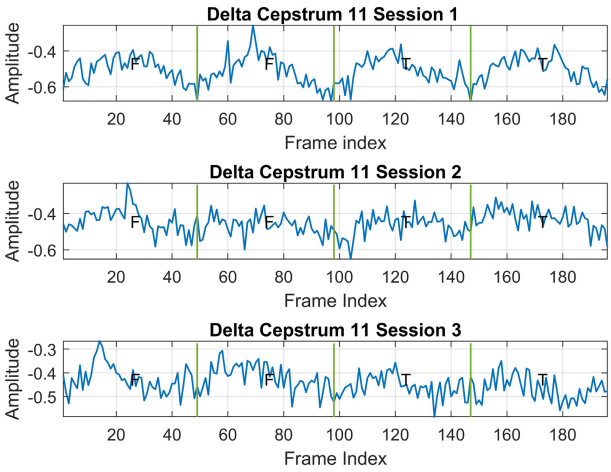

FIGURE 4. Delta cepstrum 11.

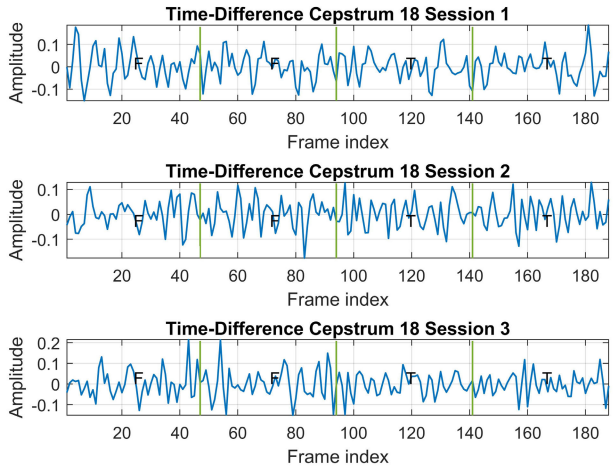

FIGURE 5. Time-difference cepstrum 18.

B. SPECTRAL FEATURE DEVELOPMENT AND EXTRACTION

Spectral energy features extracted for analysis are that of delta energy and time-difference energy feature. Each utterance was divided into frames within a 20ms duration and a 50 percent overlap for extraction. Using a 1024-point DFT, the spectral energy feature was calculated. For every frame and value, spectral energy is estimated within a certain range of frequency bins [26]. Each frame was divided into 21 Bark Bands within a 7700Hz frequency, as seen in Table 3. Spectral energy over the Bark scale is more natural in approximating the perception in the ear [18]. Critical bandwidth tends to remain constant (about 100 Hz) up to 500 Hz and increases to approximately 20 percent of the center frequency above 500 Hz [18]. The critical bandwidth [24] is calculated approximately using (4) where frequency f is in Hz.

$$BW_c(f) = 25 + 75[1 + 1.4\left(\frac{f}{1000}\right)^2]^{0.69} \quad Hz \qquad (4)$$

Frequency in Hz is converted to the Bark scale for analysis purposes. Each frame is normalized with the global masking threshold, and power spectral density to extract the spectral

TABLE 3. Spectral feature critical band filter bank.

Band Index	Critical Bandwidth(Hz)	Band Index	Critical Bandwidth(Hz)
1	DF-100	13	1720-2000
2	100-200	14	2000-2320
3	200-300	15	2320-2700
4	300-400	16	2700-3150
5	400-510	17	3150-3700
6	510-630	18	3700-4400
7	630-770	19	4400-5300
8	770-920	20	5300-6400
9	920-1080	21	6400-7700
10	1080-1270	22	7700-9500
11	1270-1480	23	9500-12000
12	1480-1720	24	12000-15500

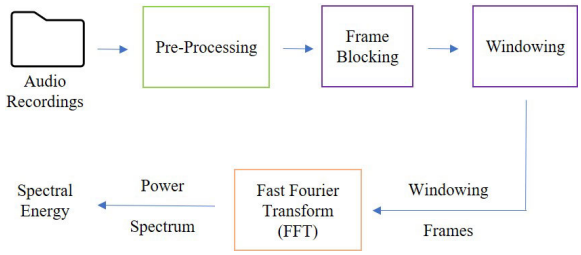

FIGURE 6. Spectral feature extraction pipeline.

energy feature [18]. For reference purposes, the quiet threshold for hearing is obtained. The total energy of the spectral components that are at least 3dB above the global threshold of every frame of utterance is calculated to get the features. To obtain the spectral energy feature specifically, the sum of each of the 21 bands' energy levels above the threshold was produced. It was then normalized to calculate the delta energy feature and the time-difference energy feature. Fig. 6 shows the initial pipeline of the spectral energy feature extraction process.

To indicate the time-dependent variation, the frame-to-frame difference energy features are extracted using the expression in (5), where i is indicative of the number of the frame index.

$$\text{difference energy (m)} = \text{NSPE}_\text{n}\,(\text{i}+1) - \text{NSPE}_\text{n}\,(\text{i}) \qquad (5)$$

The delta energy features are obtained by calculating the difference between successive normalized spectral energies' as can be seen in (6).

$$\text{delta energy (n)} = \text{NSPE}_{\text{n}+1} - \text{NSPE}_\text{n} \qquad (6)$$

Using time-difference spectrum and delta spectrum features from Bark band 12 and Bark band 1 shown as an example in Fig. 7 and Fig. 8 respectively to depict their feature domain representations of truthful and deceptive utterances of the word 'No'. Distinguishable patterns between truthful and deceptive utterance 'No' are visible.

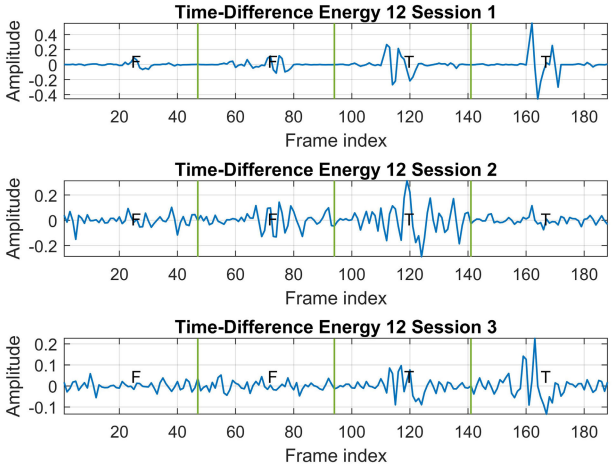

FIGURE 7. Time-difference energy 12.

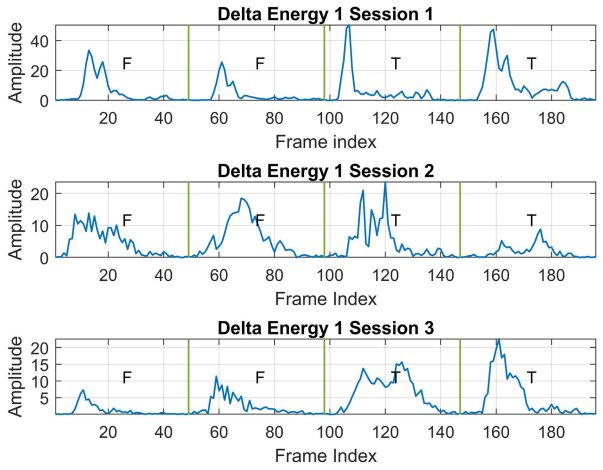

FIGURE 8. Delta energy 1.

V. PRINCIPAL COMPONENT ANALYSIS

The time-difference cepstrum and spectral feature and the delta cepstrum and spectral feature could provide the more accurate feature matching classification results by incorporating with the PCA. While working on data deception, it is observed that large datasets are often challenging to interpret the results more accurately. Therefore, the PCA aids in data dimension compression while minimizing loss of information and increasing interpretability. It is a type of reduction method that considers the original dataset as rows signifying characteristics in high dimensional space and all the rows are put up to directions that represent the best set of features [5]. The PCA constructs a group of new latent variables that reduce the dimensions of the original data space. The main variation information is then extracted from the new mapping space and extracts the statistical features. As a result, the new solution of the spatial features of the original data can be constructed. In the new mapping space, the variables are composed of linear combinations of the original data which is a method that reduces the dimensions of the projection space.

Due to the statistical eigenvectors in the projection space being orthogonal to each other, the correlation between variables is eliminated and the complexity of the original process internal function and documentation of coeff = pca(X), the principal characteristic analysis is simplified [25]. Using MATLAB's internal function and documentation of coeff = pca(X), the principal component analysis of the raw data was obtained for the n-by-p data matrix dependent on the specified feature. The function takes a feature matrix as input, in this case, the cepstrum and spectral features matrix data, and performs PCA analysis on it. Each column in the PCA coefficient matrix contains a coefficient for one principal component. The columns are organized in descending order of component variance. Using the data obtained after applying the PCA in MATLAB, Levenberg-Marquardt and LSTM feature matching techniques were used to find the recognition rate and compare the results between these two methods with existing works.

VI. FEATURE MATCHING

In many speech processing tasks, deep neural networks have been successfully used in speaker verification [27], [28], speech enhancement [29]–[31], and speech recognition [32]–[34], deception detection [6], [18], and emotion recognition [13]. Spectral and cepstrum features and the target result of 'True' and 'False' speech were applied to a Levenberg-Marquardt neural network model and an LSTM neural network model for training. The Levenberg – Marquart backpropagation algorithm and LSTM algorithm are feature matching techniques that are used to test the accuracy of training and testing data both before and after the application of PCA. Applying the PCA in the spectral energy and cepstrum feature improves the results of both the feature matching techniques. The neural network models use the extracted features to calculate a more accurate level of deception detection.

A. LEVENBERG-MARQUARDT

Levenberg-Marquardt is known for having high accuracy and speed for feed-forward neural networks [6]. The Levenberg - Marquardt algorithm works by performing a combined training process using complex curvature around the area [18]. The steepest descent algorithm is then used until the proper local curvature is acquired to make a quadratic approximation [18]. Then to speed up the convergence process, it becomes the Gauss-Newton algorithm, which means it requires less memory allocation and provides ease of use when selected. This neural network model is comprised of an input layer, multiple hidden layers, and an output layer as can be seen in Fig. 9.

Each set of spectral energy and cepstrum features is used independently in the network. Batch mode was used to train the network with a target value of 1 for deceptive speech, and 0 for nondeceptive speech. In each case, any resulting value above 0.5 was deceptive speech, and any value below 0.5 was nondeceptive speech. When training the network, it was observed that between 7 and 20 epochs were enough to achieve low mean-squared error as can be observed in Fig. 9.

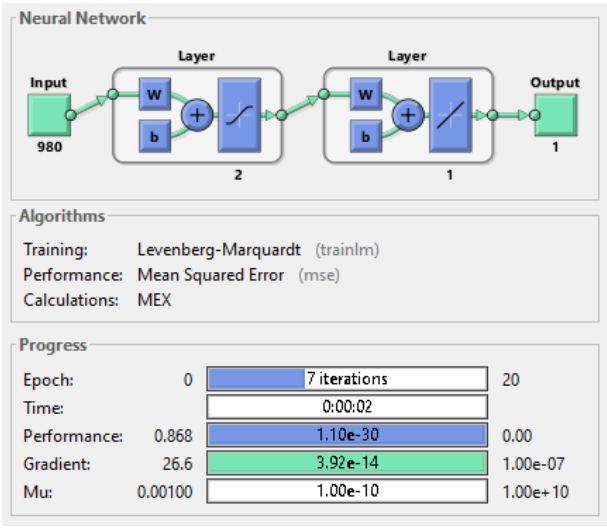

FIGURE 9. Levenberg-marquardt neural network model.

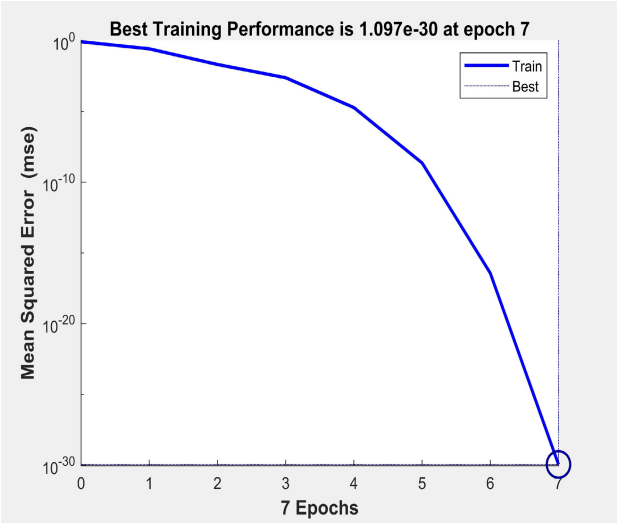

FIGURE 10. Levenberg-marquardt algorithm learning curve.

For each of the four features, Session 1 was used for training with Session 2, and Session 3 was used independently for testing. Session 2 was then used for training with Session 1, and Session 3 was used for testing. Lastly, Session 3 was used for training while Session 1, and Session 2 were used independently for testing. Table 4 presents the results before and after applying PCA for both sets of features. The trained network performed well for both sets of features especially after applying PCA. The results in Table 4 with bold red coloring are the indication of misclassified data.

The time-difference spectral energy feature correctly detected deceptive speech in 19 out of 24 (79.16%) cases before applying PCA. After applying PCA, the time-difference spectral energy feature correctly detected deceptive speech in all 24 (100%) cases. The delta spectral energy feature correctly detected deceptive speech in 18 out of 24 (75%) cases before applying PCA. After applying PCA, the delta spectral energy feature correctly detected deceptive speech in 21 out of 24 (87.5%) cases. The time-difference cepstrum feature correctly detected deceptive speech in 20 out of 24 (83.33%) cases before applying PCA. The delta cepstrum feature correctly detected deceptive speech in 19 out of 24 (79.16%) cases before applying PCA. After applying PCA, the delta spectral energy feature correctly detected deceptive speech in 22 out of 24 (91.66 %) cases.

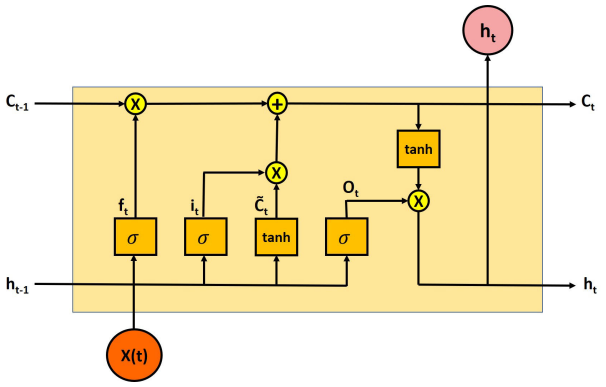

FIGURE 11. Long short-term memory network architecture.

B. LONG SHORT-TERM MEMORY

Deep learning methods like LSTM are also widely known as powerful machine learning tools for classification problems. LSTM model is a type of recurrent neural network (RNN) model that is capable of learning long-term dependencies between time-series data [20]. The LSTM neural network is an improved algorithm with neurons that can keep memory in their channels to effectively mitigate the vanishing gradient problem [23]. The key of the LSTM neural network is the cell state which can add or remove information to the cell state through the gates [35]. An LSTM network consists of four interactive cells including an input gate, an output gate, a forget gate, and an internal unit. The input gate controls the level at which the cell state updates. The output gate controls the level at which the cell state is added to the hidden state. The forget gate controls the level at which the cell state forgets of rests. It also retains information from the internal state that allows the LSTM unit to forget the unit's memory [20]. The internal unit adds information to the cell state. The gates work together to trace the flow of data from the input end to the output end of the cell. The basic internal LSTM network architecture is represented in Fig. 11.

The input gate is represented by i_t, the forget gate is represented by f_t, the internal cell candidate is represented by g_t, and the output gate is represented by o_t. The inputs are the cell state C_{t-1}, the hidden layer output h_{t-1}, and the sequence vector $X(t)$ at time t and the outputs are the cell state C_t, the LSTM hidden layer outputs h_t [22], [23].

Equation (7) is used to calculate the input gate.

$$i_t = \sigma\left(W_i \bullet [h_t - 1, x_t] + b_i\right) \tag{7}$$

TABLE 4. Feature matching results using levenberg-marquardt before and after applying PCA.

Recording Session	Utterances 'NO'	Recognition Rate (%)															
		Before PCA								*After PCA*							
		Spectral Features				Cepstrum Features				Spectral Features				Cepstrum Features			
		Time Difference		Delta		Time Difference		Delta		Time Difference		Delta		Time Difference		Delta	
		Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test
Session – 01 (Train Session 02)	Q4 – False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
	Q5 – False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
	Q7 – True	True	True	*True*	*False*	True	True	True	True	True	True	True	True	True	True	True	True
	Q9 – True	True	True	*True*	*False*	True	True	*True*	*False*	True	True	True	True	True	True	*True*	*False*
Session – 02 (Train Session 01)	Q4 – False	False	False	False	False	*False*	*True*	False	False	False	False	*False*	*True*	False	False	False	False
	Q5 – False	False	False	False	False	False	False	False	False	False	False	False	False	*False*	*True*	False	False
	Q7 – True	True	True	True	True	True	True	True	True	True	True	True	True	True	True	True	True
	Q9 – True	*True*	*False*	True	True	True	True	True	True	True	True	True	True	True	True	*True*	*False*
Session – 03 (Train Session 01)	Q4 – False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
	Q5 – False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
	Q7 – True	*True*	*False*	True	True	True	True	True	True	True	True	True	True	*True*	*False*	True	True
	Q9 – True	True	True	True	True	*True*	*False*	True	True	True	True	True	True	True	True	True	True

TABLE 5. Feature matching results using LSTM before and after applying PCA.

Recording Session	Utterances 'NO'	Recognition Rate (%)															
		Before PCA								*After PCA*							
		Spectral Features				Cepstrum Features				Spectral Features				Cepstrum Features			
		Time Difference		Delta		Time Difference		Delta		Time Difference		Delta		Time Difference		Delta	
		Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test	Actual	Test
Session – 01 (Train Session 02 & Session 03)	Q4 – False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
	Q5 – False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
	Q7 – True	True	True	True	True	*True*	*False*	*True*	*False*	True	True	*True*	*False*	True	True	*True*	*False*
	Q9 – True	True	True	*True*	*False*	True	True	*True*	*False*	True	True	True	True	True	True	True	True
Session – 02 (Train Session 01 & Session 03)	Q4 – False	False	False	False	False	False	False	*False*	*True*	False	False	False	False	False	False	*False*	*True*
	Q5 – False	*False*	*True*	*False*	*True*	False	False	False	False	False	False	False	False	False	False	False	False
	Q7 – True	True	True	*True*	*False*	True	True	True	True	True	True	True	True	True	True	True	True
	Q9 – True	True	True	*True*	*False*	True	True	True	True	True	True	*True*	*False*	True	True	True	True
Session – 03 (Train Session 01 & Session 02)	Q4 – False	False	False	False	False	False	False	*False*	*True*	False	False	False	False	False	False	*False*	*True*
	Q5 – False	False	False	False	False	*False*	*True*	*False*	*True*	False	False	False	False	*False*	*True*	False	False
	Q7 – True	True	True	True	True	True	True	*True*	*False*	True	True	True	True	True	True	True	True
	Q9 – True	True	True	*True*	*False*	True	True	True	True	True	True	True	True	True	True	True	True

Equation (8) is used to calculate the forget gate.

$$f_t = \sigma(W_f \bullet [\mathrm{h}_t - 1, \mathrm{x}_t] + \mathrm{b}_f) \tag{8}$$

To calculate the output gate, equation (9) is used.

$$O_t = \sigma(W_o \bullet [\mathrm{h}_t - 1, \mathrm{x}_t] + \mathrm{b}_o) \tag{9}$$

To calculate the output of the hidden layer, equation (10) is used.

$$h_t = O_t \bullet \tan(C_t) \tag{10}$$

To calculate the current candidate cell state, equation (11) is used.

$$\tilde{C}_t = \tanh\left(W_c \bullet [h_t - 1, x_t] + b_C\right) \tag{11}$$

To calculate the current state cell, equation (12) is used.

$$C_t = f_t \bullet C_{t-1} + i_t \bullet \tilde{C}_t \tag{12}$$

From Equation (7) – (12), σ representative of the sigmoid activation function, W_i, W_f, and W_o are representative of the input gate weight matrix, the forget gate weight matrix, and the output gate weight matrix respectively, and b_i, b_f, b_o, are representative of the input gate bias, the forget gate bias, and the output gate bias, respectively.

Each set of spectral energy and cepstrum features is used independently in the network. Batch mode was used to train the network with a target value of 1 for deceptive speech, and 2 for nondeceptive speech. When training the network, it was observed that the elapsed computational time was approximately 40-47 seconds for both the cepstrum and spectral energy features. They all ran for the max amount of 2000 epochs at 1 iteration per epoch. In each case, any resulting value below 1 was deceptive speech, and any value above 1 and below 2 was nondeceptive speech. For each of the four features, Session 1 and Session 2 were used for training, and Session 3 was used for testing. Session 2 and Session 3 were then used for training with Session 1 was used for testing. Lastly, Session 1 and Session 3 were used for training while Session 2 was used for testing. Table 5 presents the results before and after applying PCA for both sets of features.

The trained network performed well for both sets of features especially after applying PCA. The results shown in Table 5 with bold red color are the indication of misclassified data. The time-difference spectral energy feature correctly detected deceptive speech in 11 out of 12 (91.66%) cases before applying PCA. After applying PCA, the time-difference spectral energy feature correctly detected deceptive speech in all 12 (100%) cases. On the other hand, the delta spectral energy feature correctly detected deceptive speech in 7 out of 12 (58.3%) cases before applying PCA. After applying PCA, the delta spectral energy feature correctly detected deceptive speech in 10 out of the 12 (83.33%) cases. The time-difference cepstrum

TABLE 6. Results comparison of recent deception detection techniques.

Ref	Features	Classification Method	Computational Time	Recognition Rate
[2]	Short-time Energy (STE), Pitch, Format, and Duration features	Gradient Boosting Decision Tree (GBDT)	~	68.02%
[14]	The Prosodic features, Nonlinear Linear Dynamic features	RBFNN	~	42.13%
	The Prosodic features, Nonlinear Linear Dynamic features	SVM	0.2351s	68.14%
	The Prosodic features, Nonlinear Linear Dynamic features	RVM	0.0031s	70.37%
[18]	Bark Energy	Levenberg-Marquardt	3-20 epochs	83.33%
	Significant Energy	Levenberg-Marquardt	3-20 epochs	66.67%
[36]	MFCC (Pre-PCA)	Levenberg-Marquardt	3 – 7 epochs	61.33%
[40]	LexRNN: Lexical features	LSTM	~	81.03%
	HybridRNN: Speaker-dependent features and lexical features	LSTM	~	84.11%
	AudioRNN: Speaker-dependent acoustic features	LSTM	~	62.59%
[43]	Fundamental Frequency, Short-Term Energy, Zero-Crossing Rate, and MFCC	SVM	~	82.47%
This Work	Time Difference Energy	Levenberg-Marquardt	5-18 epochs	100%
	Time Difference Energy	LSTM	2000 epochs 40-47sec	100%
	Delta Energy	Levenberg-Marquardt	5-18 epochs	87.5%
	Delta Energy	LSTM	2000 epochs 40-47sec	83.33%
	Time Difference Cepstrum	Levenberg-Marquardt	5-18 epochs	83.33%
	Time Difference Cepstrum	LSTM	2000 epochs 40-47sec	91.66%
	Delta Cepstrum	Levenberg-Marquardt	5-18 epochs	91.7%
	Delta Cepstrum	LSTM	2000 epochs 40-47sec	75%

feature correctly detected deceptive speech in 10 out of 12 (83.33%) cases before applying PCA. After applying PCA, the time-difference cepstrum feature correctly detected deceptive speech in 11 out of the 12 (91.66%) cases. The delta cepstrum feature correctly detected deceptive speech in 6 out of 12 (50%) cases before applying PCA. After applying PCA, the delta cepstrum feature correctly detected deceptive speech in 9 out of the 12 (75%) cases.

When we experimented, we ran the algorithm and we trained and tested it multiple times using LSTM and Levenberg-Marquardt. Each time we ran it, we got the same results (i.e. 100%) for the time-difference energy feature. Due to the size of the data used, overfitting could have occurred. However, if we expand the size of the corpus for further analysis and verification of the test results of our proposed extracted feature using the same process and the classifier methods, the recognition rate could change from 100%.

VII. CONCLUSION AND FUTURE RESEARCH

We have extracted a few unique speech signal features. Different classifiers are applied for deception detection before and after PCA which was performed on the data. Using the Levenberg-Marquardt algorithm, it shows the correct detection of deceptive speech using the delta energy feature before applying PCA (75% recognition rate) compare to after applying PCA (87.5% recognition rate). However, the time-difference energy feature correctly detects the deceptive speech at the recognition rate of 79.17% and 100% before and after applying PCA respectively. Similarly, before applying PCA to the delta cepstrum, the deception detection rate is 79.17% and after applying PCA, the deception detection rate is at 91.7% accurate. While using the LSTM method with and without incorporation PCA in the time-difference energy, the deception detection rate is 91.7% and 100%. From the analysis, it can be inferred that the spectral energy features after applying PCA provided the best classification results. Further research using more speech utterances from a multitude of speakers, which is hard to obtain, can confirm the results of the proposed feature classification methods in detecting deception. Studying the use of a field-programmable gate array (FPGA) with the post PCA data is an implementation method where this research could expand as further research direction to create a software-hardware device to improve the accuracy for real-life applications.

REFERENCES

[1] Z. Labibah, M. Nasrun, and C. Setianingsih, "Lie detector with the analysis of the change of diameter pupil and the eye movement use method Gabor wavelet transform and decision tree," in *Proc. IEEE Int. Conf. Internet Things Intell. Syst. (IOTAIS)*, Bali, Indonesia, Nov. 2018, pp. 214–220.

[2] C. Fan, H. Zhao, X. Chen, X. Fan, and S. Chen, "Distinguishing deception from non-deception in Chinese speech," in *Proc. 6th Int. Conf. Intell. Control Inf. Process. (ICICIP)*, Wuhan, China, Nov. 2015, pp. 268–273.

[3] B. A. Rajoub and R. Zwiggelaar, "Thermal facial analysis for deception detection," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 6, pp. 1015–1023, Jun. 2014.

[4] K. Gopalan and S. Wenndt, "Speech analysis using modulation-based features for detecting deception," in *Proc. 15th Int. Conf. Digit. Signal Process.*, Cardiff, U.K., Jul. 2007, pp. 619–622.

[5] H. Roopa and T. Asha, "A linear model based on principal component analysis for disease prediction," *IEEE Access*, vol. 7, pp. 105314–105318, Jul. 2019.

[6] S. V. Fernandes and M. S. Ullah, "Phychoacoustic masking of delta and time-difference cepstrum features for deception detection," in *Proc. 11th IEEE Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, New York, NY, USA, Oct. 2020, pp. 0213–0217.

[7] I. Pavlidis and J. Levine, "Thermal facial screening for deception detection," in *Proc. 2nd Joint 24th Annu. Conf. Annu. Fall Meeting Biomed. Eng. Soc.*, Houston, TX, USA, Oct. 2002, pp. 1143–1144.

[8] D. Barsever, S. Singh, and E. Neftci, "Building a better lie detector with BERT: The difference between truth and lies," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Glasgow, U.K., Jul. 2020, pp. 1–7.

[9] R. H. Nugroho, M. Nasrun, and C. Setianingsih, "Lie detector with pupil dilation and eye blinks using Hough transform and frame difference method with fuzzy logic," in *Proc. Int. Conf. Control, Electron., Renew. Energy Commun. (ICCREC)*, Yogyakarta, Indonesia, Sep. 2017, pp. 40–45.

[10] M. H. Yap, B. Rajoub, H. Ugail, and R. Zwiggelaar, "Visual cues of facial behaviour in deception detection," in *Proc. IEEE Int. Conf. Comput. Appl. Ind. Electron. (ICCAIE)*, Penang, Malaysia, Dec. 2011, pp. 294–299.

[11] G. Tsechpenakis, D. Metaxas, M. Adkins, J. Kruse, J. K. Burgoon, M. L. Jensen, T. Meservy, D. P. Twitchell, A. Deokar, and J. F. Nunamaker, "HMM-based deception recognition from visual cues," in *Proc. IEEE Int. Conf. Multimedia Expo*, Amsterdam, The Netherlands, Jul. 2005, pp. 824–827.

[12] M. Graciarena, E. Shriberg, A. Stolcke, F. Enos, J. Hirschberg, and S. Kajarekar, "Combining prosodic lexical and cepstral systems for deceptive speech detection," in *Proc. IEEE Int. Conf. Acoust. Speed Signal Process.*, May 2006, pp. 1033–1036.

[13] M. Sanaullah and M. H. Chowdhury, "Neural network based classification of stressed speech using nonlinear spectral and cepstral features," in *Proc. IEEE 12th Int. New Circuits Syst. Conf. (NEWCAS)*, Trois-Riveres, QC, Canada, Jun. 2014, pp. 33–36.

[14] Y. Zhou, H. Zhao, X. Pan, and L. Shang, "Deception detecting from speech signal using relevance vector machine and non-linear dynamics features," *Neurocomputing*, vol. 151, pp. 1042–1052, Mar. 2015.

[15] Y. Xie, R. Liang, H. Tao, Y. Zhu, and L. Zhao, "Convolutional bidirectional long short-term memory for deception detection with acoustic features," *IEEE Access*, vol. 6, pp. 76527–76534, Nov. 2018.

[16] R. C. Cosetl and J. M. D. B. Lopez, "Voice stress detection: A method for stress analysis detecting fluctuations on lippold microtremor spectrum using FFT," in *Proc. 21st Int. Conf. Electr. Commun. Comput. (CONIELECOMP)*, San Andres Cholula, Mexico, Feb. 2011, pp. 184–189.

[17] K. Kumar, C. Kim, and R. M. Stern, "Delta-spectral cepstral coefficients for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Prague, Czech Republic, May 2011, pp. 4784–4787.

[18] M. Sanaullah and K. Gopalan, "Deception detection in speech using bark band and perceptually significant energy features," in *Proc. IEEE 56th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Columbus, OH, USA, Aug. 2013, pp. 1212–1215.

[19] A. Ghosal and S. Dutta, "Automatic male-female voice discrimination," in *Proc. Int. Conf. Issues Challenges Intell. Comput. Techn. (ICICT)*, Ghaziabad, India, Feb. 2014, pp. 731–735.

[20] J. Li, H. Chen, T. Zhou, and X. Li, "Tailings pond risk prediction using long short-term memory networks," *IEEE Access*, vol. 7, pp. 182527–182537, Dec. 2019.

[21] S. H. Rafi, N. A. Masood, S. R. Deeba, and E. Hossain, "A short-term load forecasting method using integrated CNN and LSTM network," *IEEE Access*, vol. 9, pp. 32436–32448, Feb. 2021.

[22] H. Zhou, Y. Zhang, L. Yang, Q. Liu, K. Yan, and Y. Du, "Short-term photovoltaic power forecasting based on long short term memory neural network and attention mechanism," *IEEE Access*, vol. 7, pp. 78063–78074, Jun. 2019.

[23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[24] T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proc. IEEE*, vol. 88, no. 4, pp. 451–513, Apr. 2000.

[25] Y. Zhang, B. Zhang, and Z. Wu, "Multi-model modeling of CFB boiler bed temperature system based on principal component analysis," *IEEE Access*, vol. 8, pp. 389–399, Dec. 2020.

[26] J. S. Park, J. S. Yoon, Y. H. Seo, and G. J. Jang, "Spectral energy based voice activity detection for real-time voice interface," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 17, pp. 4305–4312, Sep. 2017.

[27] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 4052–4056.

[28] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Proc. 22nd Adv. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, Dec. 2009, pp. 1096–1104.

[29] Z. Yang, J. Lei, K. Fan, and Y. Lai, "Keyword extraction by entropy difference between the intrinsic and extrinsic mode," *Phys. A, Stat. Mech. Appl.*, vol. 392, no. 19, pp. 4523–4531, Oct. 2013.
[30] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, Jan. 2014.
[31] S. Gholami-Boroujeny, A. Fallatah, B. P. Heffernan, and H. R. Dajani, "Neural network-based adaptive noise cancellation for enhancement of speech auditory brainstem responses," *Signal, Image Video Process.*, vol. 10, no. 2, pp. 389–395, Feb. 2016.
[32] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.
[33] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 30–42, Jan. 2012.
[34] J. Lei, C. Zhang, Y. Fang, Z. Gu, N. Ling, and C. Hou, "Depth sensation enhancement for multiple virtual view rendering," *IEEE Trans. Multimedia*, vol. 17, no. 4, pp. 457–469, Apr. 2015.
[35] D. Chen, J. Zhang, and S. Jiang, "Forecasting the short-term metro ridership with seasonal and trend decomposition using loess and LSTM neural networks," *IEEE Access*, vol. 8, pp. 91181–91187, May 2020.
[36] M. Sanaullah and K. Gopalan, "Distinguishing deceptive speech from truthful speech using MFCC," in *Proc. 7th Int. Conf. Circuits Syst. Signals (WSEAS)*, Cambridge, MA, USA, Feb. 2013, pp. 167–171.
[37] A. Xue, H. Rohde, and P. A. Finkelstein. (2019). *An Acoustic Automated lie Detector*. Princeton University. [Online]. Available: https://www.cs.princeton.edu/sites/default/files/alice_xue_spring_2019.pdf
[38] W. Wang, X. Yu, Y. H. Wang, and R. Swaminathan, "Audio fingerprint based on spectral flux for audio retrieval," in *Proc. Int. Conf. Audio, Lang. Image Process.*, Shanghai, China, Jul. 2012, pp. 1104–1107.
[39] J. Hirschberg, S. Benus, J. M. Brenier, F. Enos, S. Friedman, S. Gilman, C. Girand, M. Graciarena, A. Kathol, L. Michaelis, B. Pellom, E. Shriberg, and A. Stolcke, "Distinguishing deceptive from non-deceptive speech," in *Proc. 9th Eur. Conf. Speech Commun. Technol. (INTERSPEECH)*, Lisbon, Portugal, Sep. 2005, pp. 1833–1836.
[40] S. Desai, M. Siegelman, and Z. Maurer. (2017). *Neural lie Detection With the CSC Deceptive Speech Dataset*. Standford University. [Online]. Available: https://web.stanford.edu/class/cs224s/project/reports_2017/Shloka_Desai.pdf
[41] F. Enos, E. Shriberg, M. Graciarena, J. Hirschberg, and A. Stolcke, "Detecting deception using critical segments," in *Proc. 8th Annu. Conf. Int. Speech Commun. Assoc. (ISCA)*, Antwerp, Belgium, Aug. 2007, pp. 2281–2284.
[42] T. Warnita and D. P. Lestari, "Construction and analysis of Indonesian-interviews deception corpus," in *Proc. 20th Conf. Oriental Chapter Int. Coordinating Committee Speech Databases Speech I/O Syst. Assessment (O-COCOSDA)*, Seoul, South Korea, Nov. 2017, pp. 1–6.
[43] H. Tao, P. Lei, M. Wang, J. Wang, and H. Fu, "Speech deception detection algorithm based on SVM and acoustic features," in *Proc. IEEE 7th Int. Conf. Comput. Sci. Netw. Technol. (ICCSNT)*, Daliam, China, Oct. 2019, pp. 31–33.
[44] E. Wong and S. Sridharan, "Comparison of linear prediction cepstrum coefficients and mel-frequency cepstrum coefficients for language identification," in *Proc. Int. Symp. Intell. Multimedia, Video Speech Process. (ISIMP)*, Hong Kong, May 2001, pp. 95–98.

SINEAD V. FERNANDES (Graduate Student Member, IEEE) received the B.S. degree in computer engineering from Florida Polytechnic University (FPU), Lakeland, FL, USA, in 2016, where she is currently pursuing the M.Sc. degree in electrical engineering, under the supervision of Dr. Muhammad Sana Ullah. Her research interests include deception detection and speech signal processing. She is one of the Founding Member of the IEEE HKN-Mu Omega chapter with the Florida Polytechnic University.

MUHAMMAD S. ULLAH (Senior Member, IEEE) received the B.S. degree in electrical and electronic engineering from the Chittagong University of Engineering and Technology, Chittagong, Bangladesh, in 2008, the M.S. degree in electrical and computer engineering from Purdue University Northwest, Hammond, IN, USA, in 2013, and the Ph.D. degree in electrical and computer engineering from the University of Missouri-Kansas City, Kansas City, MO, USA, under the supervision of Dr. M. Chowdhury.

He joined the College of Engineering, Florida Polytechnic University, Lakeland, FL, USA, as an Assistant Professor of computer engineering. His current research interests include a relatively new methodology and nanotechnology for the next generation of computing and other micro- and nano-electronic applications; modeling of RLC interconnects and RF interconnect in high-density integrated circuits (ICs); investigation of a tunneling device based on transition metal dichalcogenide material, graphene, carbon nanotube, and other emerging 2-D nanomaterials; very large scale integration signal processing; and higher-order statistics and spectra in signal processing.