Hindawi
International Journal of Digital Multimedia Broadcasting
Volume 2020, Article ID 8927031, 10 pages
https://doi.org/10.1155/2020/8927031

Research Article

Identification of Weakly Pitch-Shifted Voice Based on Convolutional Neural Network

Yongchao Ye,[1] **Lingjie Lao**,[1] **Diqun Yan**,[1,2] **and Rangding Wang**[1]

[1]*Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China*
[2]*Guangdong Key Laboratory of Intelligent Information Processing and Shenzhen Key Laboratory of Media Security, Shenzhen 518060, China*

Correspondence should be addressed to Diqun Yan; yandiqun@nbu.edu.cn

Received 3 June 2019; Revised 12 August 2019; Accepted 22 August 2019; Published 6 January 2020

Academic Editor: Yifeng He

Pitch shifting is a common voice editing technique in which the original pitch of a digital voice is raised or lowered. It is likely to be abused by the malicious attacker to conceal his/her true identity. Existing forensic detection methods are no longer effective for weakly pitch-shifted voice. In this paper, we proposed a convolutional neural network (CNN) to detect not only strongly pitch-shifted voice but also weakly pitch-shifted voice of which the shifting factor is less than ±4 semitones. Specifically, linear frequency cepstral coefficients (LFCC) computed from power spectrums are considered and their dynamic coefficients are extracted as the discriminative features. And the CNN model is carefully designed with particular attention to the input feature map, the activation function and the network topology. We evaluated the algorithm on voices from two datasets with three pitch shifting software. Extensive results show that the algorithm achieves high detection rates for both binary and multiple classifications.

1. Introduction

Voice disguising [1] is commonly used in forensic scenario as an effective mean of concealing the identity of the speaker. And it can be divided into two categories, nonelectronic disguising and electronic disguising. Nonelectronic disguising voice is usually obtained by pinching the nose, covering the mouth, pulling the check, etc., which is easy to be noticed under human supervision. Electronic disguising is achieved by using electronic devices or software to modify the voice pitch and format.

The simplest way of electronic disguising is to change the playback speed of the target voice. Although the speaker's identity could be concealed, the rhythm of the disguised voice generated in this way is relatively unnatural and is not often adopted by the attackers in practice. Pitch shifting is a typical electronic disguising technique in which the pitch of the voice is changed while keeping the duration unchanged. Generally, the pitch-shifted voice is more natural in terms of timbre, tone, etc., and difficult to be detected. In this paper, we mainly focus on identification of pitch-shifted voices.

Clark [2] studied the ability of the human to distinguish the electronic disguised voice, and quantitatively analyzed the different effects of the different pitched voice on the human hearing. Wu et al. [3–5] studied the mechanism of pitch shifting and constructed a pitch shifting dataset with various voice software/tools. The final detection accuracy of their method can reach up to 90%, while keeps the false alarm rate less than 10%. However, the performance on weakly pitch-shifted voices is relatively poor. Especially for the voices shifted with ±4 semitones, the detection rates drop lower than 90%. In [6], environment noise is considered in identifying the pitch shifting. The experimental results show that the features extracted from linear frequency cepstral coefficients (LFCC) and formant can be effectively discriminant the natural and pitch-shifted voice. However, the experimental results on weakly pitch-shifted voices have not been given in [6].

Recently, some studies on the detection of weakly pitch-shifted voices have been reported. Based on [5], Liang et al. [7] focused on voice with the shifting factors of ±4 semitones, but the promotion is limited. Singh [8] compared performance of different classifiers on the voice shifted with semitones from ±2 to ±10. However, the result performed on a dataset with dozens of voice samples is not consistent.

Convolutional Neural Networks (CNN) [9] have achieved state-of-the-art performance in computer vision, data mining,

as well as automatic speaker verification. And CNN have been adopted to audio forensics as well [10, 11]. Chen et al. [12] identified various audio post processing operations by a CNN. Especially for small size voice samples, the network achieves significant improvement comparing with other works. In [13], unlike other hand-crafted features, a CNN is adopted to capture the steganographic modifications adaptively and outperform the traditional methods.

Although many methods have been proposed for pitch shifting identification, there is still room to improve the performance especially when the suspected voices are weakly-shifted. In this paper, a CNN model for pitch shifting detection is proposed. By analyzing the principle of voice pitch shifting, LFCC and the first derivative coefficients are used as identification features. Comparing to other related works, the proposed CNN achieves remarkable performance in both binary and multiple classifications. The main contributions of our work are summarized as follows.

(i) High accuracy is achieved on identifying weakly pitch-shifted voice. Since the difference between the original voice and the weakly pitch-shifted voice is little, the identification is a challenging task in previous work.

(ii) Utilizing CNN architecture to identify the pitch-shifting voice, which improve the performance compared to the previous work. And the proposed network architecture is carefully devised.

(iii) Massive experiments are conducted on two dataset and three pitch shifting software, which indicates the proposed method achieved great robustness.

The remainder of the paper is organized as follows. In Section 2, we briefly introduce the principle of voice pitch shifting. Section 3 presents the identification features and describes the proposed CNN topology. In Section 4, a series of experiment results are given. Finally, the paper is concluded in Section 5.

2. Voice Pitch Shifting

Voice pitch shifting can be performed in either time-domain or frequency domain. Time-domain Pitch Synchronous Overlap Add (TD-PSOLA) is a commonly used approach which works by windowing [14]. Upsampling achieves pitch shifting by moving the segments further apart and downsampling achieves by moving closer together. Upsampling can achieve the compression of the spectrum, which lowers the pitch. Downsampling can achieve the expansion of the spectrum, thus raise the pitch. In real scenarios, more state-of-art voice synthesis algorithms are applied in audio editing software. These algorithms have better performance in timbre and rhythm. In our work, Audition [15], GoldWave [16] and Audacity [17] are considered as pitch shifting methods.

In this paper, we use semitone to measure the pitch of shifted voice. A semitone is the smallest interval between two tones. It is defined as the interval between two adjacent notes in a 12-tone scale [18], which means the frequency between two adjacent semitones has an equal ratio of $2^{1/12}$. In other words, if the voice frequency is raised or lowered by $2^{1/12}$ times, the pitch can be raised or lowered by one semitone. Let f_0 be the frequency of original voice, and the frequency of pitch-shifted voice f is given by the following formula

$$f = f_0 \times 2^{m/12}, \quad m = \pm 1, \pm 2, \ldots, \pm 11, \tag{1}$$

where m represents the semitones of pitch-shifted voice compared to original one. A positive m means raising the pitch of voice and a negative one means lowering the pitch of voice. In this paper, we use m as a shifting factor which denotes the pitch-shifted voice.

3. Identification Algorithm Based on CNN

3.1. Feature Extraction. We randomly choose a voice sample from the TIMIT [19] dataset and shift the voice by setting m in Equation (1) to −4 and +4 respectively. The waveform and spectrogram of original and pitch-shifted voice are shown in Figure 1. As we can see, the shifting operation changes the waveform little while leaves traces on the frequency domain. Thus, acoustic features which characterize frequency domain can be applied to the proposed algorithm.

LFCC is a cepstral feature widely used in voice identification and achieves significantly performance [20]. Recent works [21] show that LFCC can more effectively captures the lower as well as higher frequency characteristics than other cepstral coefficients. Hence, in this work, LFCC is considered to extract the identification feature. The extraction procedure of LFCC is as follows.

The voice signal is first pre-processed with pre-emphasized and then windowed. Let $s(n)$ be the preprocessed voice signal and $n = 0, 1, \ldots, N-1$, where N is the duration of the signal. Suppose the frequency spectrum $S_i(k)$ of the i-th voice frame is calculated by short-time Fourier transform (STFT), k refers to the k-th spectrum. Then the power spectrum filtered by a set of linearly-spaced triangular filters can be defined by

$$P_i(l) = \sum_{k=0}^{N-1} \left[S_i(k)\right]^2 F_l(k), \quad 0 \le l < L, 0 < i < M, \tag{2}$$

where L is the number of filters and M is the number of frames in a voice sample. $F_l(k)$ is defined as

$$F_l(k) = \begin{cases} \dfrac{k - o(l)}{c(l) - o(l)} & o(l) \le k \le c(l), \\ \dfrac{h(l) - k}{h(l) - c(l)} & c(l) \le k \le h(l), \end{cases} \tag{3}$$

where $o(l)$, $c(l)$ and $h(l)$ are the lowest frequency, central frequency, and the highest frequency of l-th filter, respectively. The adjacent filters have $c(l) = h(l-1) = o(l+1)$.

Finally, the DCT is applied to the Log-power of the L filters to calculate the LFCC of $s(n)$

$$LFCC_i(j) = \sqrt{\frac{2}{L}} \sum_{l=0}^{L-1} \log[P_i(l)] \cos\left(\frac{\pi n(2l-1)}{2L}\right), \tag{4}$$

where $LFCC_i(j)$ is LFCC of i-th frame, and j is the index of DCT coefficients.

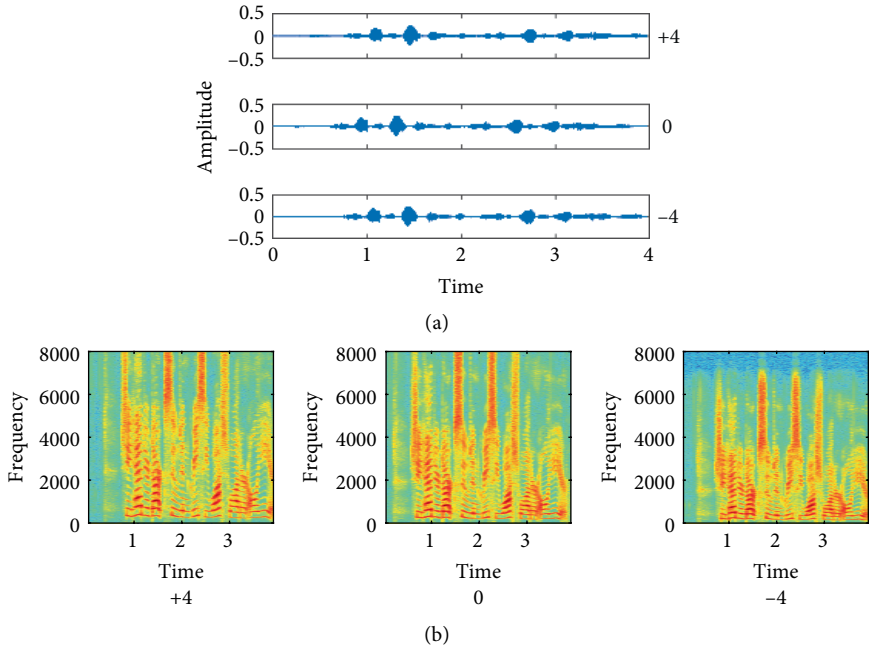

Figure 1: Waveform and spectrogram of original voice and pitch-shifted voice. (a) Waveform; (b) Spectrogram.

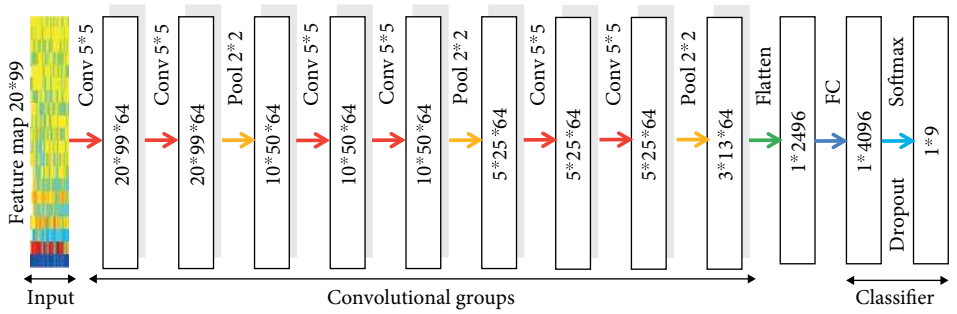

Figure 2: Proposed CNN architecture.

Since most of the pitch shifting techniques do not fully model temporal characteristics of voice [22], the dynamic coefficients, such as the first and second derivatives, could be useful in identifying pitch-shifted voice. In this work, we take the first derivative into consideration and it could be given by

$$\Delta LFCC_i = \frac{\sum_{n=1}^{N} n\left(LFCC_{i+n} - LFCC_{i-n}\right)}{2\sum_{n=1}^{N} n^2}. \quad (5)$$

$\Delta LFCC_i$ is the first derivative coefficient of i-th frame, which computed in term of the static coefficients $LFCC_{i+n}$ to $LFCC_{i-n}$. A typical value for N is 2.

3.2. Proposed CNN Architecture

3.2.1. Network Topology. Convolution neural networks have shown remarkable performance in various classification tasks. It generally consists of an input layer, multiple hidden layers and an output layer. The hidden layers are crucial to the network performance, which typically are combination of different kinds of layers such as convolutional layers, pooling layers and full connected layers [9].

The proposed network architecture is shown in Figure 2. The input of the network is the $\Delta LFCC$ matrix, and the output is a predict label, which indicates the suspected voice is pitch shifted or not. The entire network consists of three convolutional groups, a fully connected layer and a softmax layer. In the training stage, after extracting features of voice segments, the $\Delta LFCC$ feature matrix is fed into the network. The specific size of matrix depends on length of each frame and number of filters. Then it undergoes three convolutional groups which are stacked one after another. Next, the feature map of the last convolutional group is fed into the fully connected layer. All the weight values in the network will be updated via back propagation. The testing stage is mostly as same as the training stage. The $\Delta LFCC$ feature matrix of the suspected voice is first extracted and undergoes the whole network. A softmax is used as the classifier at the end of network.

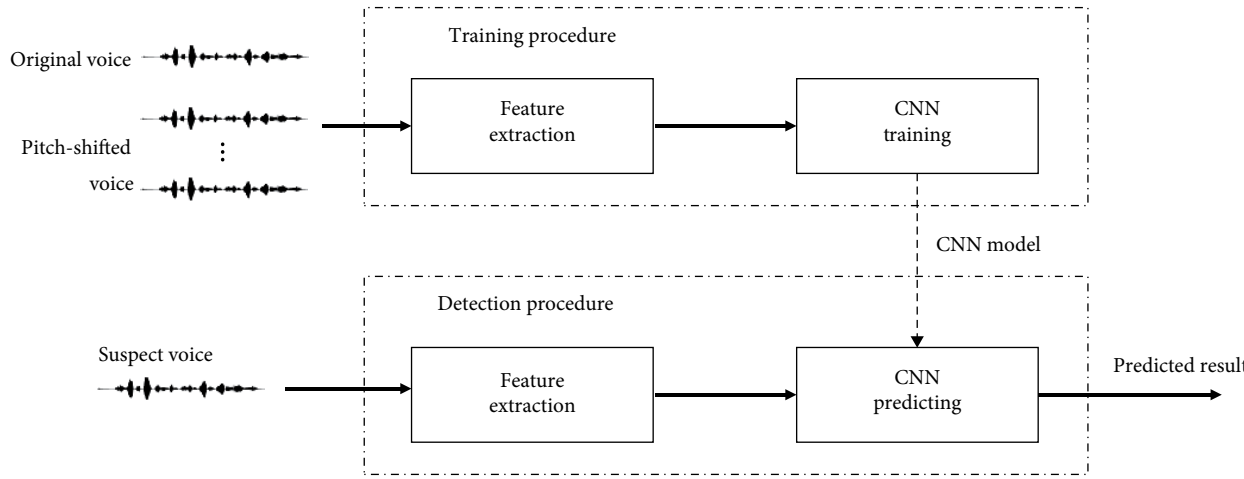

Figure 3: Diagram of the proposed pitch-shifting identification algorithm.

3.2.2. Convolutional Group. In our network, each convolutional group includes two convolutional layers and a pooling layer. The convolution layer consists of a set of linear convolutional filters which can generate local feature maps. Two-dimensional convolutional layer preforms a convolution on the input feature map with a specific kernel size. Let x_i^{l-1} be the input feature map of the i-th neuron at layer $l-1$, output feature map is computed as

$$y_j^l = \sum_i \omega_{j,i}^l \cdot x_i^{l-1}, \tag{6}$$

where y_j^l is the output map of the j-th neuron at layer l, and $\omega_{j,i}^l$ is the weight value between the j-th neuron at layer l and the i-th neuron at the previous layer $l-1$. All convolutional layers use the same kernel size and number of stride (5 * 5 size, 1 * 1 stride). Since the *ΔLFCC* feature map is a two-dimensional matrix, the first convolutional layer in the first group has one input channel and 64 output channels, while the other convolutional layers have both input channels and output channels with number of 64. Nonlinear activation functions can enhance the mapping capacity of the model by introducing nonlinearity into the network.

Pooling layers are adopted after convolutional layers which can obtain more global information by combining the feature information extracted from the convolution layer. Max pooling is commonly used in the pooling layer. It is a downsampling operation, which chooses the maximum value within a local window is taken as the output

$$y_j^l = \max x_j^{l-1}, \quad x_j^{l-1} \in X, \tag{7}$$

where X is the pooling region in feature map. The region is defined by pool size and number of strides. Pooling layers reduce the number of parameters in the network significantly and have little effect on input feature map, thus decrease the computational cost and prevent over-fitting. All max-pooling layers use the same pool size and number of stride (2 * 2 size, 2 * 2 stride).

3.2.3. Rest Part of Network. After three convolutional groups, the fully connected layer acts as a "classification" map in the network, which can do the high-level reasoning and learn distributed feature representation. Neurons in fully connected (FC) layer are connected to all activation functions in the previous layer. However, overly complex networks will reduce the generalization of the model. Dropout is a simple and effective regularization technique to prevent over-fitting [23]. Hence, in our network, we drop out half of input neurons in the FC layer.

Table 1: Architecture and parameters of the proposed network.

No.	Layer	Kernel size/ neuron numbers	Strides	Input channels	Parameters
1	Convolutional 1	(5,5)	(1,1)	1	1664
2	Convolutional 2	(5,5)	(1,1)	64	102464
3	Pooling 1	(2,2)	(2,2)	64	—
4	Convolutional 3	(5,5)	(1,1)	64	102464
5	Convolutional 4	(5,5)	(1,1)	64	102464
6	Pooling 2	(2,2)	(2,2)	64	—
7	Convolutional 5	(5,5)	(1,1)	64	102464
8	Convolutional 6	(5,5)	(1,1)	64	102464
9	Pooling 3	(2,2)	(2,2)	64	—
10	Flatten	2496	—	—	—
11	Fully connected	4096	—	—	1.02 * 107
12	Softmax	N[1]	—	—	4096 * N

[1] N depends on specific the number of classes.

Softmax can be considered as an effective multiple-output competitive whose output represents the likelihood of classification. Therefore, the dimension of its output represents the number of classes. Let N be the number of classes, the probabilities of input data over N different classes are predicted by the softmax function

$$p(z)_j = \frac{e^{z_j}}{\sum_{n=1}^{N} e^{z_n}} \quad j = 1, \ldots, N, \tag{8}$$

where z_j is the output of the FC layer on each class. Finally, the predicted label depends on the largest probability max$p$.

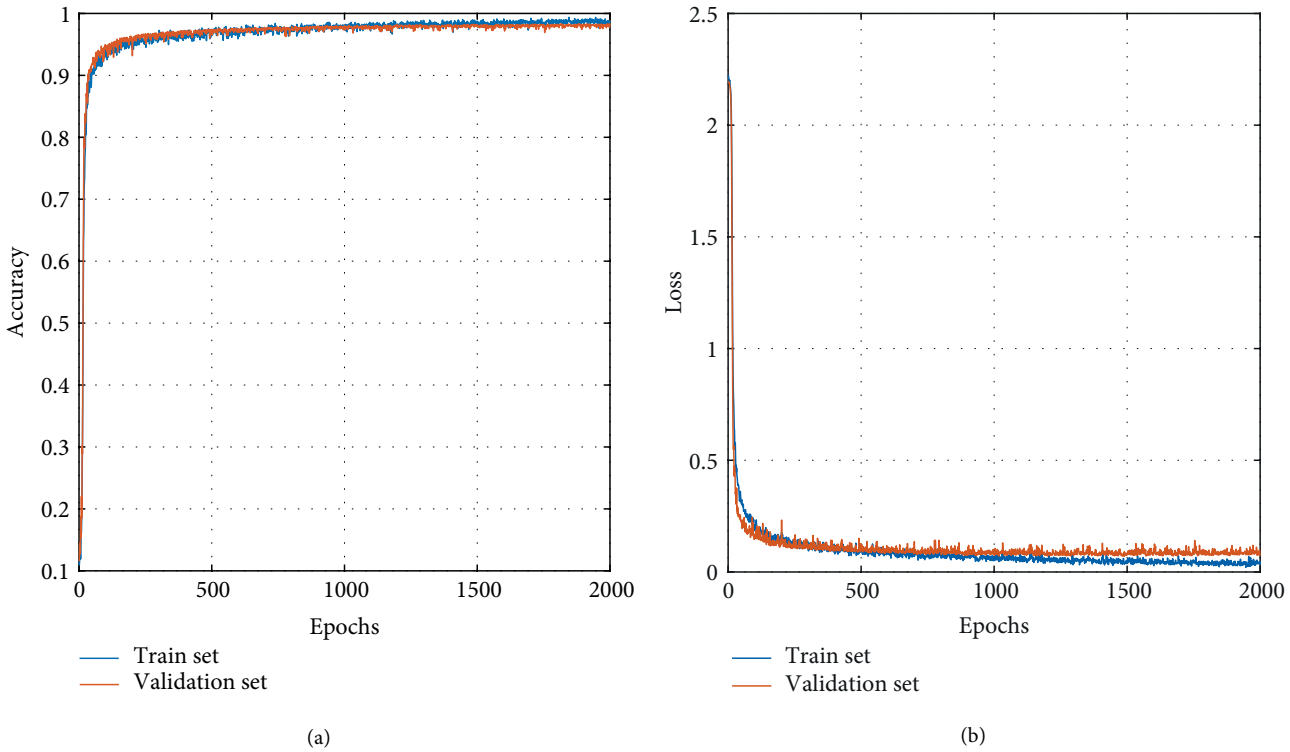

Figure 4: The training process of proposed network.

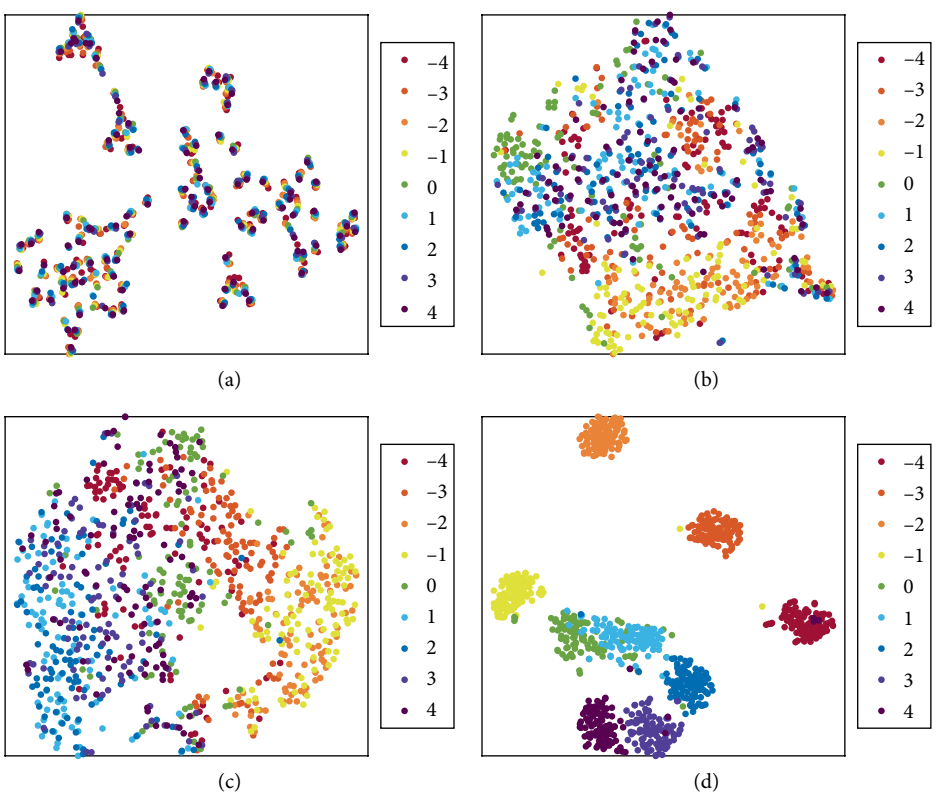

Figure 5: Visualization of different feature maps by *t*-SNE. (a) The first derivative of LFCC. (b) Output feature map of first Conv group in CNN. (b) Output feature map of second Conv group in CNN. (c) Output feature map of last Conv group in CNN.

Table 2: Detection performance of strongly pitch-shifted voice in binary classification.

Pitch shifting software	Training dataset	Testing dataset	Detecting method					
			[6] LFCC + GMM		[8] MFCC + GMM		Proposed	
			Rate	FAR	Rate	FAR	Rate	FAR
Audition	TIMIT	TIMIT	99.86	0.02	**99.88**	**0.02**	99.54	0.10
	TIMIT	UME	97.60	**1.10**	**98.06**	1.19	95.89	1.52
	UME	TIMIT	**99.52**	0.36	98.58	**0.02**	97.51	1.45
	UME	UME	**99.79**	0.15	**99.79**	**0.12**	99.49	**0.12**
GoldWave	TIMIT	TIMIT	**99.97**	**0.00**	99.94	0.01	99.58	0.05
	TIMIT	UME	**97.93**	**0.75**	96.82	2.04	96.29	1.53
	UME	TIMIT	**99.72**	0.05	98.45	**0.01**	98.44	1.17
	UME	UME	**99.87**	**0.02**	99.70	0.07	99.12	0.36
Audacity	TIMIT	TIMIT	**99.98**	**0.00**	99.97	**0.00**	99.97	**0.00**
	TIMIT	UME	99.13	0.44	97.57	2.10	**99.78**	**0.07**
	UME	TIMIT	**99.97**	0.01	98.72	**0.00**	99.96	0.01
	UME	UME	**99.97**	**0.00**	99.95	**0.00**	99.84	0.11

Table 3: Detection performance of weakly pitch-shifted voice in binary classification.

Pitch shifting software	Training dataset	Testing dataset	Detecting method					
			[6] LFCC + GMM		[8] MFCC + GMM		Proposed	
			Rate	FAR	Rate	FAR	Rate	FAR
Audition	TIMIT	TIMIT	98.11	0.83	97.29	1.34	**98.72**	**0.70**
	TIMIT	UME	92.95	5.50	93.25	**1.67**	**96.83**	1.84
	UME	TIMIT	96.72	**0.47**	95.21	1.72	**97.26**	0.52
	UME	UME	97.70	0.88	**97.82**	**0.64**	96.82	0.91
GoldWave	TIMIT	TIMIT	97.92	0.68	**98.93**	**0.42**	98.14	1.47
	TIMIT	UME	82.86	14.60	91.56	**4.64**	**92.98**	5.95
	UME	TIMIT	92.58	**0.13**	93.93	0.25	**96.84**	1.25
	UME	UME	98.39	**0.08**	**98.78**	0.14	97.79	0.92
Audacity	TIMIT	TIMIT	98.27	0.32	**99.55**	**0.06**	99.10	0.29
	TIMIT	UME	83.04	15.44	87.96	10.07	**94.25**	**4.05**
	UME	TIMIT	91.89	0.06	91.84	**0.03**	**98.12**	0.33
	UME	UME	98.89	**0.09**	**99.30**	**0.09**	98.39	0.87

In summary, the architecture and parameters of the proposed network are shown in Table 1.

3.3. Proposed Identification Algorithm for Pitch-Shifted Voice. The proposed identification algorithm is based on the first derivative of LFCC and CNN classifier. With a group of equaling distributed triangular filters, LFCC can capture more characteristics both in low frequency and high frequency comparing with other acoustics features such as MFCC. Thus, the difference between the original voice and the pitch-shifting voice are easier to be distinguished. CNN is considered to have better performance in classification task for multi-layers process with less time and subsampling layers give better feature extraction. The proposed algorithm consists of training and testing stages, as shown in Figure 3.

In the training stage, the voice pitch-shifted different factors and the original voice are considered as separate classes. After extracting the first derivative of LFCC based on Equation (5), feature map together with labels are fed into the network for training.

In the testing stage, the first derivative of LFCC are first extracted and then fed into the trained CNN model. The probability given by softmax in Equation (8) reveals the voice is more likely to be the original one or shifted with which semitone.

4. Results and Discussion

4.1. Experiment Setup. In the experiments, the proposed algorithm is evaluated on TIMIT [19] and UME [24]. TIMIT consists of 6300 voice samples from 630 speakers with the average duration of 3 s. And it is turned into three different sub-datasets using Audition, GoldWave, and Audacity respectively, each of which contains sixteen shifting factors from ±1 semitones to ±8 semitones. Hence, there are totally 100800 voice samples in each sub-dataset of TIMIT. Similarly, UME consists of 4040 voice samples from 202 speakers with the average duration of 5 s. TIMIT and UME are turned into three sub-datasets respectively, each of which composed

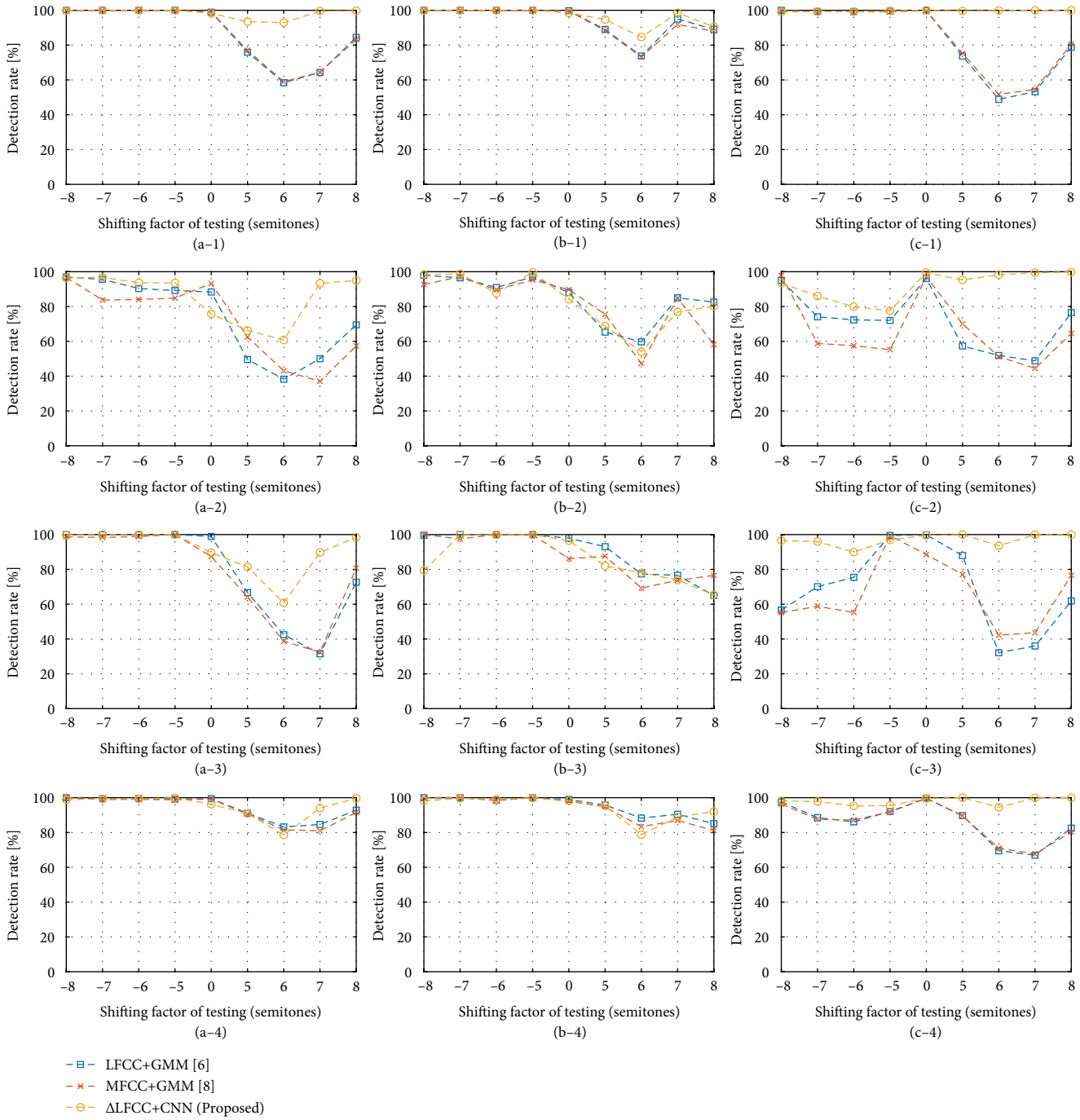

FIGURE 6: Detection rates of strongly pitch-shifted voice. (a-*x*) Voice pitch-shifted by Audition. (b-*x*) Voice pitch-shifted by GoldWave. (c-*x*) Voice pitch-shifted by Audacity; *x represents subfigures in same column*. (*y*-1) TIMIT for training and TIMIT for testing. (*y*-2) TIMIT for training and UME for testing. (*y*-3) UME for training and TIMIT for testing. (*y*-4) UME for training and UME for testing; *y represents subfigures in same row.*

of 64640 voice samples. In each sub-dataset, 60% of voice samples are selected randomly into training dataset, 20% sample into validation dataset and the remaining 20% sample into testing dataset. Speaker identity is not considered while splitting, and two datasets are from different speakers. Thus, the datasets are supposed to be speaker independent. Those voice samples with the shifting factor less than ±4 semitones are considered as weakly pitch-shifted, while others are strongly pitch-shifted. All the voice samples from both datasets are WAV, 16 KHz sampling rate, 16-bit quantization and mono.

For each voice sample, 20-dimensional LFCC feature map is extracted by setting the length of frame N to 256 and the number of filters L to 20 in Equation (2). In [6], LFCC with SVM classifier achieves great robustness detecting disguised voice in noisy environment. In our work, the GMM classifier

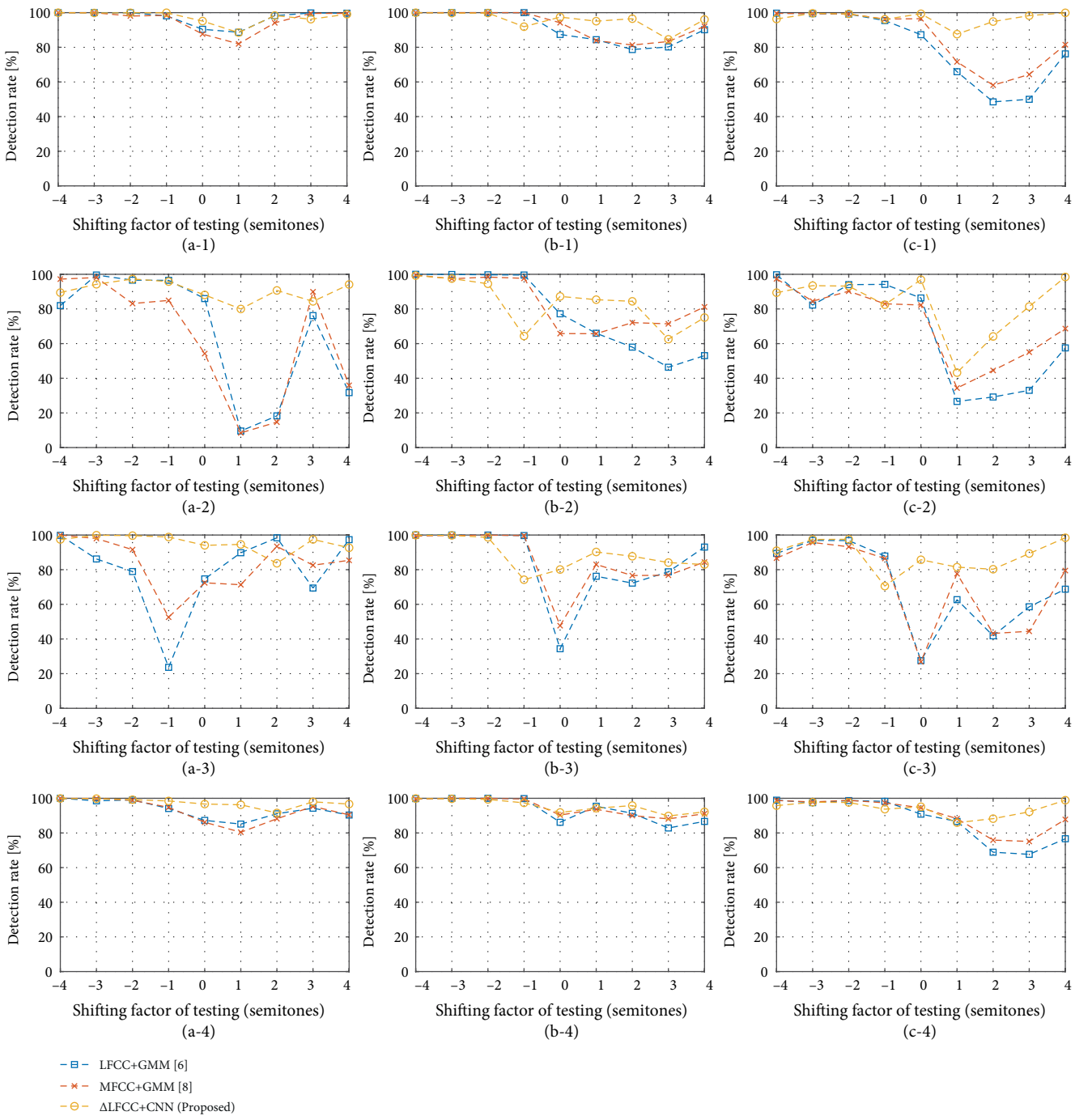

FIGURE 7: Detection rates of weakly pitch-shifted voice. (a-x) Voice pitch-shifted by Audition. (b-x) Voice pitch-shifted by GoldWave. (c-x) Voice pitch-shifted by Audacity; *x represents subfigures in same column*. (y-1) TIMIT for training and TIMIT for testing; (y-2) TIMIT for training and UME for testing. (y-3) UME for training and TIMIT for testing. (y-4) UME for training and UME for testing; *y represents subfigures in same row.*

is used as a comparison, among which the number of GMM kernels is set to 256.

The detection rate is used to evaluate the performance of the proposed network. Let N_p be the number of pitch-shifted voice samples and N_o be the number of original voice samples. Assuming that N_p' and N_o' are the voice samples from pitch-shifted voices and original voices which are identified as pitch-shifted. The detection rate is defined as N_p'/N_p. Meanwhile, a false alarm is the most serious of the voiceprint authentication system errors to some extent. Therefore, in addition to using the detection rate to assess the proposed algorithm, we also considered the False Alarm Rate (FAR) is the testing stage. The FAR is defined as N_o'/N_o.

4.2. CNN Training. In this paper, TanH is utilized as activation function in the proposed network. We use Adam algorithm [25] with an initial learning rate of 0.0001 to accelerate the training. The proposed network is trained for 2000 iterations

with the batch size of 32. The training process is presented in Figure 4, which shows the proposed network is neither overfitting nor underfitting.

t-Distributed Stochastic Neighbor Embedding (*t*-SNE) is a dimensionality reduction method which tries to place the objects in a low-dimensional space so as to optimally preserve neighbourhood identity. And it is particularly suitable for the visualization of high-dimensional data [26] such as the output feature maps of convolutional layers.

We randomly choose 100 voice samples from each sub-dataset of TIMIT which shifted with shifting factors from ±1 semitones to ±4 semitones by Audition. Each sample is fed into the trained the network respectively, and the output feature maps of convolution layers are recorded. Figure 5 shows the visualization results of four feature maps using *t*-SNE. The process from Figures 5(a) to 5(d) demonstrates the proposed network can capture the difference between the original voice and voice pitch-shifted with different factors. In Figure 5(a), all voice samples are mixed together at first, which indicates that the characteristic represented by first derivative of LFCC is more related to voice itself rather than pitch-shifting factors. In Figure 5(d), samples from same class clustered well, which indicated that the trained network can achieve both binary and multiple classifications.

4.3. Strongly Pitch-Shifted. In this case, as a comparison to [6] and [8], we focus on the voice strongly shifted with factors from ±5 to ±8 semitones. Firstly, we try to identify whether the suspected voice is original or pitch-shifted one. All the pitch-shifted voice (shifted ±5 to ±8 semitones) are taken as negative samples in binary classification. In real forensic scenarios, the pitch-shifted voice can be recorded by variety of devices in different environments. Hence, cross-dataset experiments are necessary and important. The detection rates and FARs of this case are presented in Table 2.

It can be seen that, all the detection methods achieve a detection rate higher than 95% and FAR lower than 2%. The method in [6] performs best in binary classification, for it achieves the highest detection rate and lowest FAR in most cases. Although the proposed method does not perform as well as [6] and [8], the gaps in both detection rates and FARs are less than 1%. These minor differences may have little effect on the detection performance.

Compared with binary classification, multiple classification is more practical for real forensic application. In this case, we not only recognize whether the suspected voice is pitch-shifted, but also determine the specific shifting factor. The results are presented in Figure 6. First, as we can see from Figure 6, the detection rates of voice shifted with negative factors are higher than those with positive factors. The main reason for this phenomenon is that, downsampling (raising the pitch) will amplify the spectrum which brings more noise, while upsampling compress the spectrum. Second, different pitch shifting software have an impact on detection performance. The proposed method remains generally steady while others fluctuate greatly. Finally, the detection rates drop obviously in the cross-dataset evaluation, especially for a few specific semitones are lower than 50% in [6] and [8]. And it can be seen that; the detection rates of proposed method remain higher than 60% in every case when crossing training set and testing set. Hence, for those strongly pitch-shifted voice, compared with exist methods, the proposed method achieves generally the same the performance in binary classification and show more generalization ability in multiple classification.

4.4. Weakly Pitch-Shifted. In this case, we focus on weakly pitch-shifted samples shifted from ±1 to ±4 semitones which are more challenging to detect. Like Section 4.2, the binary classification is evaluated first as using all the pitch-shifted voice as negative samples. The detection rates and FARs are shown in Table 3. Compared with those strongly pitch-shifted voice, performance of all detection methods dropped. However, unlike Table 2, the proposed method performs best in Table 3. It achieves the highest detection rate and lowest FAR in most cases. Though the performance drops a little in intra-dataset, the proposed method achieves a significant improvement in cross-dataset evaluation. The detection rates remain higher than 93% in every case while others drop lower than 88%. This phenomenon can be attributed to the factor that, both LFCC and MFCC mainly focus on the static features which are more related to the voice characteristic, while *ΔLFCC* captures dynamic features which are more related to the shifting trace.

Like the previous section, multiple classification is adopted after the binary evaluation. The result show in Figure 7 reveals the proposed method performance on weakly pitch-shifted voice form ±1 to ±4 semitones.

Generally, in Figure 7, as the same trend shown in Figure 6, raising the pitch is still difficult to detect compared with lowering the pitch. And it is noted that the fluctuation on detection rates when using different pitch shifting software is still unavoidable. The first row and the last row in Figure 7 indicate the intra-dataset results, the detection rates of proposed method are higher than 90% in most cases, while others are greatly affected by different pitch shifting software and even drop lower than 60%. The 2nd and 3rd rows show the cross-dataset results, especially for a few specific semitones, both [6] and [8] lower than 20%. Proposed methods remain a steady performance with the worst case of ~60% and ~80% for most cases.

Hence, both binary and multiple classifications show that the proposed algorithm achieves good performance and has strong robustness in detecting weakly pitch-shifted voice.

5. Conclusions

In this paper, an algorithm for pitch-shifted voice identification is proposed. A convolutional neural network architecture is designed and adopted as the classifier to detect the pitch-shifted voice while linear frequency cepstral coefficients are extracted as acoustic features. The algorithm is evaluated on two datasets and three audio editing software. Extensive results indicate that the proposed algorithm achieves much better detection rates and FARs in most cases, and the proposed network shows better generalization ability comparing to traditional classifier such as GMM. Next, network architecture which can replace handcrafted acoustic features is also one of the directions worth studying.

Data Availability

The open source databases used in this work have been listed in the reference.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded by the National Natural Science Foundation of China, grant numbers [61300055, 61672302]; Natural Science Foundation of Zhejiang, grant number [LY17F020010, LY20F020010]; Natural Science Foundation of Ningbo, grant number [2017A610123] and Zhejiang College Students Science and Technology Innovation Training Program, grant number [2018R405033].

References

[1] P. Perrot, G. Aversano, and G. Chollet, "Voice disguise and automatic detection: review and perspectives," in *Progress in Nonlinear Speech Processing (Lecture Notes in Computer Science)*, Springer-Verlag, New York, NY, USA, pp. 101–117, 2007.

[2] J. Clark and P. Foulkes, "Identification of voices in electronically disguised speech," *International Journal of Speech, Language and the Law*, vol. 14, no. 2, pp. 195–221, 2007.

[3] Y. Wang, Y. Deng, H. Wu, and J. Huang, *Blind Detection of Electronic Voice Transformation with Natural Disguise*, Springer, Berlin, Heidelberg, 2013.

[4] H. Wu, Y. Wang, and J. Huang, "Blind detection of electronic disguised voice," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3013–3017, IEEE, Vancouver, BC, Canada, 2013.

[5] H. Wu, Y. Wang, and J. Huang, "Identification of electronic disguised voices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 3, pp. 489–500, 2014.

[6] W. Cao, H. Wang, H. Zhao, Q. Qian, and S. M. Abdullahi, "Identification of electronic disguised voices in the noisy environment," in *Digital Forensics and Watermarking. IWDW 2016. Lecture Notes in Computer Science*, Y. Shi, H. Kim, F. Perez-Gonzalez, and F. Liu, Eds., vol. 10082, pp. 75-87, Springer, Cham, 2017.

[7] H. Liang, X. Lin, Q. Zhang, and X. Kang, "Recognition of spoofed voice using convolutional neural networks," in *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, IEEE, Montreal, QC, pp. 293–2972017.

[8] M. K. Singh, A. K. Singh, and N. Singh, "Multimedia analysis for disguised voice and classification efficiency," *Multimedia Tools and Applications*, pp. 1–17, 2018.

[9] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[10] H. Ali, S. N. Tran, E. Benetos, and A. S. d'Avila Garcez, "Speaker recognition with hybrid features from a deep belief network," *Neural Computing Applications*, vol. 29, no. 6, pp. 13–19, 2018.

[11] K. Sundararajan and D. L. Woodard, "Deep learning for biometrics: a survey," *ACM Computing Surveys*, vol. 51, no. 3, p. 65, 2018.

[12] B. Chen, W. Luo, and D. Luo, "Identification of audio processing operations based on convolutional neural network," in *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security — IH&MMSec'18*, ACM, New York, NY, USA, pp. 73–77, 2018.

[13] B. Chen, W. Luo, and H. Li, "Audio steganalysis with convolutional neural network," in *Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security – IHMMSec'17*, ACM, New York, NY, USA, pp. 85–90, 2017.

[14] F. Charpentier and M. Stella, "Diphone synthesis using an overlap-add technique for speech waveforms concatenation," in *ICASSP', 86. IEEE International Conference on Acoustics, Speech, and Signal Processing,* vol. 11, pp. 2015–2018, 1986.

[15] "Adobe Audition CC. A professional audio workstation" February 2019, https://www.adobe.com/products/audition.html.

[16] "GoldWave — audio editor, recorder, converter, restoration, & analysis software" February 2019, http://www.goldwave.ca/.

[17] "Audacity: free audio editor and recorder" February 2019, https://www.audacityteam.org/.

[18] S. Trehub, A. Cohen, L. Thorpe, and B. Morrongiello, "Development of the perception of musical relations: semitone and diatonic structure," *Journal of Experimental Psychology: Human Perception Performance*, vol. 12, no. 3, pp. 295–301, 1986.

[19] "Timit Acoustic-Phonetic Continuous Speech Corpus," February 2019, https://catalog.ldc.upenn.edu/LDC93S1.

[20] F. Alegre, R. Vipperla, A. Amehraye, and N. Evans, "*A new speaker verification spoofing countermeasure based on local binary patterns*", in *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association*, p. 5, 2013.

[21] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association*, pp. 2087–2091, 2015.

[22] F. Soong and A. Rosenberg, "On the use of instantaneous and transitional spectral information in speaker recognition," *IEEE Transactions on Acoustics, Speech, Signal Processing*, vol. 36, no. 6, pp. 871–879, 1988.

[23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[24] "Advanced utilization of multimedia to promote higher education reform speech database" February 2019, http://research.nii.ac.jp/src/en/UME-ERJ.html.

[25] D. P. Kingma and J. Ba, "Adam a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.

[26] L. V. D. Maaten and G. Hinton, "Visualizing data using *t*-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.