IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 26 July 2022, accepted 12 August 2022, date of publication 19 August 2022, date of current version 29 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3200479

RESEARCH ARTICLE

Voice Conversion Based Augmentation and a Hybrid CNN-LSTM Model for Improving Speaker-Independent Keyword Recognition on Limited Datasets

YESHANEW ALE WUBET AND KUANG-YOW LIAN, (Member, IEEE)
Department of Electrical Engineering, National Taipei University of Technology, Taipei 10608, Taiwan

Corresponding author: Kuang-Yow Lian (kylian@mail.ntut.edu.tw)

This work was supported by the Ministry of Science and Technology (MOST), Taiwan, under Grant 110-2221-E-027-121.

ABSTRACT Keyword recognition is the basis of speech recognition, and its application is rapidly increasing in keyword spotting, robotics, and smart home surveillance. Because of these advanced applications, improving the accuracy of keyword recognition is crucial. In this paper, we proposed voice conversion (VC) - based augmentation to increase the limited training dataset and a fusion of a convolutional neural network (CNN) and long-short term memory (LSTM) model for robust speaker-independent isolated keyword recognition. Collecting and preparing a sufficient amount of voice data for speaker-independent speech recognition is a tedious and bulky task. To overcome this, we generated new raw voices from the original voices using an auxiliary classifier conditional variational autoencoder (ACVAE) method. In this study, the main intention of voice conversion is to obtain numerous and various human-like keywords' voices that are not identical to the source and target speakers' pronunciation. Parallel VC was used to accurately maintain the linguistic content. We examined the performance of the proposed voice conversion augmentation techniques using robust deep neural network algorithms. Original training data, excluding generated voice using other data augmentation and regularization techniques, were considered as the baseline. The results showed that incorporating voice conversion augmentation into the baseline augmentation techniques and applying the CNN-LSTM model improved the accuracy of isolated keyword recognition.

INDEX TERMS CNN-LSTM, data augmentation, speaker-independent keyword recognition, voice conversion.

I. INTRODUCTION

In recent years, the use of speech for human-machine interactions and devices that support voice communication has increased rapidly owing to advancements in digital technology. Currently, most hand-based human-to-machine interactions are being replaced by computer vision and speech recognition technology in an advanced way. If the size of the training data is limited, the speaker-independent keyword spotting is a very challenging task owing to overfitting problems. Considering this difficulty, this paper focuses on voice conversion (VC)-based augmentation to increase the training dataset size for deep learning algorithms and to improve speaker-independent keyword recognition. The identification of keywords is applicable for controlling robotics, speech-to-text, home surveillance (door and TV control), military activities (air force), keyword verification (unknown keyword detection), personal digital assistance (car driver and chatbot), Google search by voice, personal virtual assistance (Siri, Google Assistant, Cortana, and Alexa), aerospace applications, keyword spotting, and security [1], [2], [3], [4], [5], [6], [7], [8]. For instance, voice-based automated smart home surveillance is useful for assisting elderly and disabled people.

VC has become very sophisticated and has many applications, such as generating new voices for text-to-speech

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

(TTS) [9], [10], hiding the identity of the speaker, music conversion [11], [12], accent conversion [13], emotion conversion [14], [15], speech enhancement [16], film industry, gaming technology, and voice restoration [17]. VC is useful for people who lose their voice organs either due to nature or disease. Challenges of VC competition have been initiated and released in recent years to improve VC performance. Three VC challenges [18], [19], [20] have been addressed to date. Traditional VC methods use Gaussian mixture models (GMM), but the converted speech quality is often degraded owing to over-smoothing. To overcome this problem, a minimum distance spectral mapping (MDSM)-based GMM has been proposed [21]. The GMM-based VC is a statistical conversion method based on the maximum-likelihood estimation of spectral parameter feature statistics [22]. Recently, researchers who participated in VC challenges used different neural network approach models, such as the encoder-decoder model (Zero-Shot Voice Style Transfer with Only Autoencoder Loss, vector quantized variational autoencoders, cyclic variational autoencoder), one-shot VC, generative adversarial network (GAN) (CycleGAN-VC, StarGAN-VC, and Adaptive GAN or AdaGAN), parallel spectral mapping (Tacotron), and one-shot VC [20].

The basic goal of VC is converting the source speaker's accent to the targeted speaker's accent accurately with the full linguistic content. A large amount of data is needed for an accurate voice conversion process. If we convert the voice accurately, it is not helpful for data augmentation because we have almost the same existing voice on the limited dataset. In our scenario, the generated voice should be the modified accent of the existing voices with the full linguistic content. Many state-of-the-art VC methods [23], [24], [25] have been proposed and implemented for parallel and non-parallel VC. It is possible to train the parallel VC in a limited dataset [26]. If the performance of VC is not precise enough, voice augmentation for VC is possible. Different augmentations techniques for VC were proposed such as attention-based speaker embeddings for one-shot VC and data augmentation-based non-parallel VC [27], [28].

The main contribution of this paper is applying the advanced parallel VC techniques to real applications, specifically to increase the training data size and usage of state-of-the-art machine learning algorithms for speaker-independent keyword recognition. We consider that the very high similarity between the converted voice and the existing target voices has no significant implication for data augmentation. We realized that exact VC is not useful for voice-based augmentation. The converted voice should be a modified version of the target and source speakers' pronunciation, while the linguistic content of the keyword is maintained as it is. The proposed VC is carried out across a limited number of non-native English speakers. We reduced the training time of the VC for reducing the accurate voice conversion performance. Since the VC model degrades the quality of results for never seen voices, the trained data has been fed to the trained model during the conversion phase to simplify the challenge of the huge training data demands of VC. These techniques distinguished our approach from the VC-based augmentation of related works [36], [37], [38]. The test data of VC is already included in the training data of VC. Several speakers are not required necessarily for our VC process. Dataset-I was collected and formulated for three non-native English speakers' countries. The test data for dataset-II contained four different native language speakers, whereas the training data contained only the same native language speakers. Both dataset-I and dataset-II were organized for speaker-independent keyword recognition challenge, which is arduous relative to the speaker-dependent on limited dataset. The significance of the proposed voice augmentation technique was compared with the ordinary voice recognition augmentation and regularization techniques. Although the related works [30], [31], [33], [34] showed that the CNN model is an exemplary model for vocabulary-size speech recognition, we have proven that the fusion CNN-LSTM model is superior to the pure CNN and pure LSTM for two separate datasets. The LSTM model improved the inconsistent performance of the CNN model when CNN and LSTM were hybridized together. Since the LSTM controls the exploding gradient problems [29], we noted that replacing the fully connected layer of the CNN with LSTM reduced the vanishing gradient problem. Finally, we realized that selecting the optimal mel-spectrogram segmentation frame size values for the time distributed CNN-LSTM model has a significant impact on model performance and it needs very critical experimental investigation to achieve desirable model performance with an optimized computational time. The mel-spectrogram features are well-organized in the form of 2-D sequential frames that is very learnable and suitable for our CNN-LSTM model. A delicate CNN-LSTM framework is also designed carefully for feature extraction and classification, which could take less computation time during model training and testing. The frame size was obtained using a deep experimental analysis.

The rest of this paper is organized as follows. We described the related works in Section II. The proposed methodology is described in Section III. The dataset setup is explained in Section IV. In Section V, the results and a discussion are presented. Finally, the conclusion of this study is summarized in Section VI.

II. RELATED WORKS

Many studies have been proposed for keyword identification by applying the CNN model to mel-frequency cepstral coefficients (MFCC) and spectrogram speech signal features. Li and Zhou [30] proved that a CNN outperformed a deep feed-forward network for six-command voice recognition using MFCC feature extraction. The six commands ("up", "down", "left", "right", "unknown keyword", and "silence") were selected and used from Google's TensorFlow speech commands dataset for their experiment. Waqar *et al.* [31] proposed speech command recognition using CNN to control popular snake games. The authors used

TABLE 1. **(a) summary of related works for keyword recognition; (b) summary of related works for VC based Augmentation in limited data.**

(a)

References	Algorithm	Vocabulary size	Accuracy (%)	Training and testing data dependency
Li and Zhou [30]	CNN	6	94.5	Not pure speaker-independent
Waqar *et al.* [31]	CNN	4	96.5	Not pure speaker-independent
Wubet and Lian [32]	CNN-SVM	12	93	Not pure speaker-independent
Cayir and Navruz [33]	CNN	12	64.81	Accent-dependent on a small data
			94.64	Accent-dependent on a large data
			33.18	Accent-independent on small data
			63.29	Accent-independent on a large data
Yang *et al.* [34]	CNN	10	92.88	Not pure speaker-independent
Our proposed work	CNN-LSTM	12	94.2	Speaker-independent on limited dataset I
		10	96.8	Speaker-independent on limited dataset II

(b)

References	VC process	VC Method
Shahnawazuddin *et al.* [36]	Adults to children	CycleGAN
Singh *et al.* [37]	Adults to children	CycleGAN
Baas and Kamper [38]	Cross-linguistic	Combination of several techniques (speech encoder, style encoder, content encoder, decoding, and vocoder models)
Our proposed work	Across a limited number of non-native English speakers	ACVAE

a limited dataset for only four direction speech commands (“Up”, “Down”, “Left”, and “Right”). The MFCC features of the speech commands and the CNN algorithm were proposed to recognize these four speech commands. The experimental results showed that the proposed algorithm achieved high recognition accuracy. Similarly, Wubet and Lian [32] showed that CNN is better than the SVM model for keyword recognition, and surprisingly, a hybrid of CNN-SVM outperformed pure CNN and pure SVM. Cayir and Navruz [33] investigated the influence of a limited size dataset for voice command recognition using 12 different voice commands (“down”, “forward”, “follow”, “go”, “left”, “on”, “off”, “right”, “stop”, “up”, and “yes”). Their experimental results showed that when the test dataset included native Turkish speakers, the test accuracy was 94.64% for a large dataset and 64.81% for a small dataset. In contrast, when the test dataset included foreigners’ voices, the test accuracy declined to 63.29% for the large dataset and 33.18% for the small dataset. They examined and confirmed the above-listed results using a CNN on the MFCC features. The results indicated that the test accuracy rates increased as the training dataset size increased and the accent of the diversified voice was expanded. Yang [34] compared a speech recognition of command words performances using a deep neural network (DNN) and recurrent neural network (RNN) for 10 command voice recognition using MFCC feature extraction. The 10 commands (“yes”, “no”, “up”, “down”, “left”, “right”, “on”, “off”, “stop”, and “go”) were selected and used from Google’s TensorFlow speech commands dataset for their investigation. The result showed that CNN outperformed compared to DNN and RNN. Furthermore, Fendji *et al.* [35] have mentioned and summarized the last two decades’ study of automatic speech recognition (ASR) using limited vocabularies and sentences. Overall, most of the recently proposed models have shown that the CNN model is an exemplary model for vocabulary-size speech recognition. The comparison of the related works and the proposed model are summarized in Table 1 (a). Besides, the related works that employed VC data augmentation for speech recognition in limited data are summarized in Table 1 (b).

VC-based data augmentation has been used by several researchers in recent years. Shahnawazuddin *et al.* [36] proposed a VC-based data augmentation to improve children’s speech recognition in limited data scenarios. In this study, the acoustic attributes of adults were converted into children’s speech using a cycle-consistent GAN. Word error rates (WERs) were significantly reduced by VC-based data augmentation. However, our VC scenario does not involve exact VC processing; rather, it is the process of obtaining a human-like modified voice version of the source and target speakers’ pronunciation. Singh *et al.* [37] used VC-based data augmentation for ASR using CycleGAN and also compared its performance with the baseline system. The experimental results showed a good improvement after 200 hours of CycleGAN-based new adult speech with a reduction of 5.58% in WER compared to the baseline system. Furthermore, the collection of other augmentation and CycleGAN-converted adult speech showed the highest reduction of 7.44% in WER compared to the baseline system. Baas and Kamper [38] proposed a VC-based augmentation to improve the speech recognition system for limited data of the low-resource languages. Authors augmented the unseen and cross-linguistic low resource-limited data using a good resource language of the VC training model.

III. PROPOSED MODELS

The proposed model was developed using ACVAE-VC for voice-based data augmentation and a hybrid CNN-LSTM

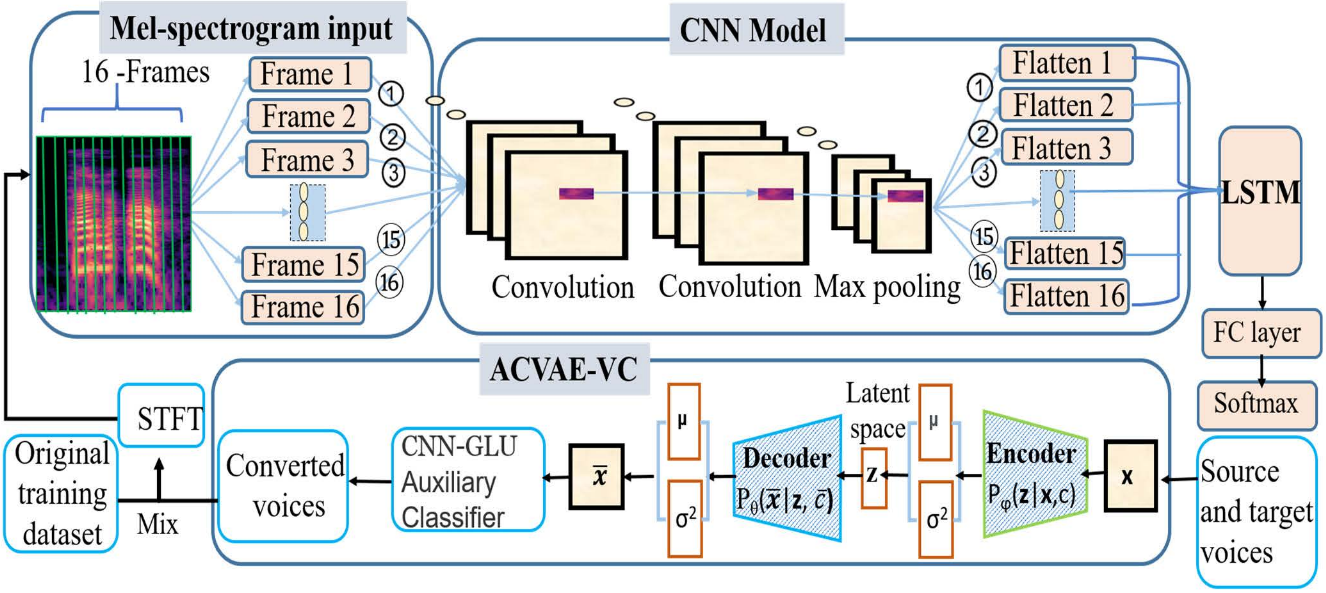

FIGURE 1. Proposed model for improving speaker-independent keyword recognition on limited datasets.

model for feature extraction and classification, as depicted in Fig. 1. The minimum mean-square error log-spectral amplitude estimator algorithm [39] is applied for noise reduction. The VC-based augmentation and a hybrid of CNN-LSTM network are briefly described as follows.

A. VOICE CONVERSION-BASED DATA AUGMENTATION

Data augmentation is the process of creating new, slightly altered samples from the original samples to escalate the training set, which can be regarded as a type of regularization method. Geometric transformation (affine transformation), generative adversarial networks (GAN), and autoencoder networks are common methods for generating more spectrogram images to reduce the overfitting problems of speech recognition in all machine learning algorithms. In addition, dropout, batch normalization, transfer learning, and one-shot learning are regularization techniques and exceptionally common ways of reducing the overfitting problem in deep neural networks [40]. Most research has shown that affine transformation has significantly improved the performance of overfitted models when compared to others. We considered geometric transformation, batch normalization, and dropout as the baseline for comparison with the proposed augmentation technique.

We proposed the ACVAE VC model [41] for VC processing. Although Kameoka *et al.* [41] used ACVAE for non-parallel VC and they aimed to generate exact accent translation on phrases and sentence utterances, we prepared and used the keyword dataset for parallel VC to keep the linguistic content of voices perfectly and to obtain a moderately modified version of target and source speakers' accents. We noted that VC-based voice augmentation should not be the exact pronunciation conversion, but the linguistic content of the keywords should be accurately maintained. Our work is speaker-independent keyword recognition (test data is completely from never seen speakers) and the number of speakers is limited. This limited number of training speakers leads to a limited dataset size for speaker-independent keyword recognition and an overfitting problem. Therefore, we need to diversify the training speakers' accents to make them look like many different speakers. Consider that we have a limited number of speakers (A and B) who speak each keyword several times, as we specified in Section IV. VC among these speakers is possible to generate new artificial speakers D and E. Speaker A to B conversion yields speaker D, whereas B to A conversion gives speaker E.

The proposed VC model uses a sequence of mel-cepstral coefficients computed from a spectral envelope sequence obtained using WORLD [42]. In the autoencoder model, the encoder network generates a set of parameters (mean and variance) for the conditional distribution $P_{\phi}(\boldsymbol{z}|\boldsymbol{x})$ of a latent space variable $\boldsymbol{z}$ from the input data $\boldsymbol{x}$, whereas the decoder network generates a set of parameters (mean and variance) for the conditional distribution $P_{\theta}(\boldsymbol{x}|\boldsymbol{z})$ of data $\boldsymbol{x}$ from the latent space variable $\boldsymbol{z}$. In regular CVAEs, the encoder and decoder are free to ignore c by finding distributions satisfying $P_{\varphi}(\boldsymbol{z}|\boldsymbol{x}, c) = P_{\varphi}(\boldsymbol{z}|\boldsymbol{x})$ and $P_{\theta}(\bar{\boldsymbol{x}}|\boldsymbol{z}, \bar{c}) = P_{\theta}(\bar{\boldsymbol{x}}|\boldsymbol{z})$. Class category c can be represented as a single one-hot vector identification of classes in ACVAE. A gated linear unit (GLU)-based CNN auxiliary classifier was introduced and applied next to the decoder to avoid VAEs problems. The classifier predicted the attribute classes of the decoder outputs [41].

As we mentioned in the introduction Section, we have found that the exact accent translation is not useful for our work because our target is acquiring various human-like voices and keeping the linguistic content for raising the number of training data. To get these successfully modified voice versions, the maximum iteration of exact VC was reduced.

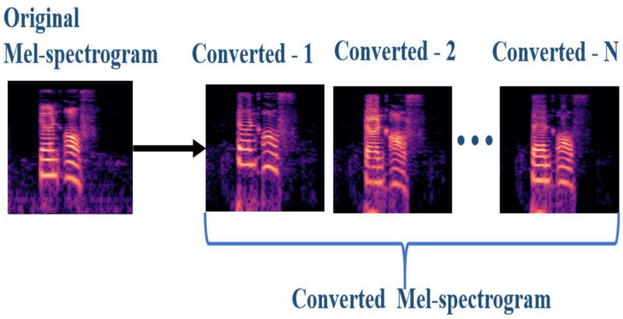

FIGURE 2. VC augmentation samples.

The noise has been reduced before and after the VC process. If there is any noise in raw data, it highly affects the VC quality. In this work, the principal goal of VC is for generating a human-like (natural) and a variant voice. Therefore, we didn't give attention to exact VC processing rather gave more caution to acquire the natural voice and to keep the linguistic content. Since our target is not designing an accurate model for VC, we used the trained data on the ACVAE training model to produce various human-like keyword voices. It was implemented across non-native English speakers' accents each other for all datasets. We considered females to females, females to males, males to males, and males to females VC to get a more generalized voice and to reduce the test data accent variation which is a big tackle of speaker-independent speech recognition. All selected speakers are both source and target speakers for ACVAE, but the same speaker cannot be both source and target at the same time. Finally, the converted voices and original voices are mixed and converted to a 2-D spectrogram by using a short-time Fourier transform (STFT) with a 23 milliseconds frame size and at a sample rate of 16000 Hz. The sample mel-spectrogram image result of VC-based augmentation is depicted in Fig. 2.

B. PROPOSED HYBRID CNN-LSTM MODEL

The proposed feature extraction and classification model was developed using a fusion of the CNN and LSTM models. Although the pure CNN model has demonstrated exceptional achievements in many applications [43], [44], [45], [46], the LSTM is integrated into it to achieve a good performance. The 2D CNNs have been proposed for the extraction of deep features of spectrograms. The 2D mel-spectrogram was split into an equal-size sequence of 16 frames. All sequences of frames are still 2D mel-spectrogram features and they are fed to 2D-CNNs sequentially based on the labeled sequence number in Fig. 3. First, frame 1 was fed into the CNN, which extracted the basic features of frame 1 and generated a flattened vector called Flatten 1. Next, frame 2 was fed into the CNN, which extracted the basic features of frame 2 and generated a flattened vector called Flatten 2. Similarly, all remaining frames were fed into the CNN, and flattened vectors were generated based on their sequence order. Although the original dimension of the mel-spectrogram was large, it was resized to 64×64×3 to reduce the computational time and space. We verified that reducing the original size to 64×64×3 pixels had no significant impact on the accuracy metrics. As a result, all 16 frames were 64×4×3 in size.

After many inspection methods to find an appropriate CNN architecture, we have found that a CNN architecture with two convolutional layers and one max-pooling layer was performed expertly. Because the segmented frame length was short, it was not possible to use many convolution layers preceding the max-pooling layers. Max-pooling layers were placed after the two convolutional layers to downsample the convolution dimension. A rectified linear unit (ReLU) activation function was applied between the convolution layers. In addition to VC–based data augmentation and affine transformation, batch normalization and dropout regularization were applied to pure CNN, pure LSTM, and CNN-LSTM to prevent the models from overfitting problems and for convenient comparison.

In this study, we fused a state-of-the-art LSTM deep learning algorithm with the sequentially flattened layer of the CNN. LSTM is employed for deep feature extraction and classification. It has shown advanced performance for sequential data prediction and classification in many applications [47], [48], [49], such as time series trend forecasting, image classification, speech classification, and sentiment analysis. Similarly, hybrid CNN and LSTM models [50], [51], [52] have improved pure CNN and pure LSTM models.

The LSTM consists of operations, activation functions, and states for receiving inputs over time. At each time step, an input vector is fed into the LSTM. We used LSTM for global temporal information extraction and classification using the extracted features of the CNNs. In a fully connected (FC) layer of LSTM, a softmax activation function was applied and used as the classifier. The CNN extracted features are carefully organized time series data for the LSTM input time series data for the LSTM input. The flattened vectors of the CNN are fed to the LSTM with 16 time steps, as depicted in Fig. 3. The CNN flattened vector output $V_{(t=i)}$ was assigned to the CNN's frame i input, where t is the time step, and i is the frame number. Each flattened vector is fed to the interconnected LSTM networks as $\boldsymbol{x}_i$ at t = i.

LSTM consists of an input gate, a forget gate, and an output gate, which are represented by i_t, f_t and o_t, respectively. It has a cell state (c) and a hidden state (h), which are the long-term memory and short-term memory, respectively. In the LSTM gates, σ is the element-wise sigmoid function and tanh is the element-wise tangent activation function. The LSTM gates processed the flattened input vector ($\boldsymbol{x}_t \in R^{N\times 1}$) at t time-step with the previous short-term memory (h_{t-1}), where $N \times 1$ is the size of vectors. Finally, the new cell state and the new short-term memory are computed according to:

$$c_t = f_t \odot c_{t-1} + i_t \odot \hat{c}_t \tag{1}$$

$$h_t = o_t \odot tanh(c_t) \tag{2}$$

where c_{t-1} is the previous cell state, $\hat{c}_t$ is the candidate cell state, c_t is the new cell state, and $\odot$ is an element-wise product operator.

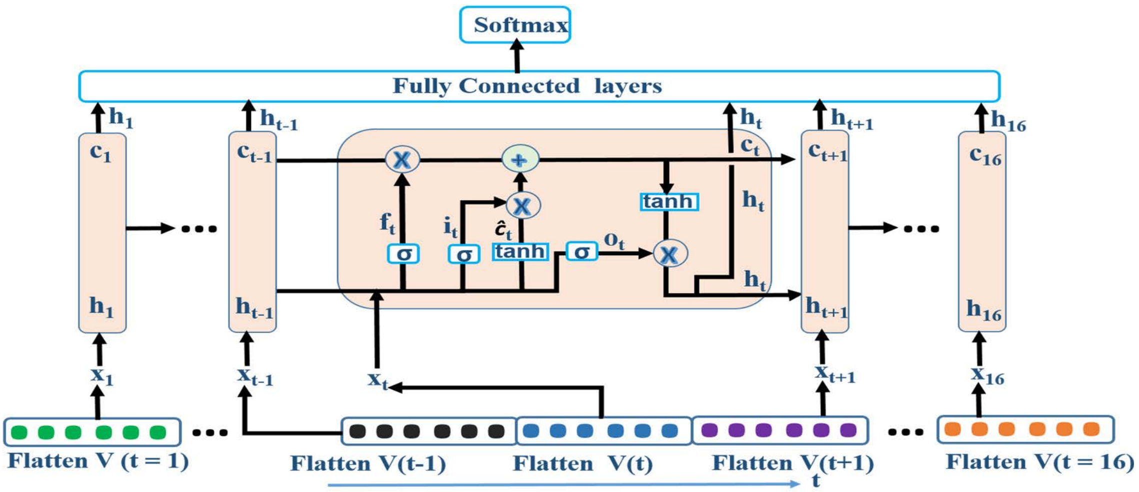

FIGURE 3. Proposed model for improving speaker-independent keyword recognition on limited datasets.

The spectrogram input was segmented in the vertical direction, but the horizontal segmentation was not as good as the vertical direction. When the number of frames is high, the frame length is too small. Conversely, when the number of frames is low, the frame lengths are large. Selecting a large frame length is preferred for a good trainable CNN model, whereas many segmented frames are required for a good learnable LSTM model. However, the CNN-LSTM model requires a good adjustment of the frame size to design a worthy model. We simplified the complication by selecting an optimal and more generalized 2^i number of frames per single spectrogram and frame length for a 64×64×3 spectrogram image:

$$Number\ of\ Frames = 2^i \tag{3}$$

$$Frame\ length = \frac{L}{2^i} \tag{4}$$

where $i = 0, 1, 2, \ldots, 6$, and L is the length of the mel-spectrogram. Because frame segmentation was applied vertically on each spectrogram, the frame width was kept the same as the spectrogram width. We compared the performance among all 2^i frames experimentally, and we realized that a frame size of 16×64×4×3 per single spectrogram adjustment is surprisingly the best CNN-LSTM input from other frame sizes. From the selected frame size, 16 is the number of frames, 4 is the width of the frames, 64 is the height of the frames, and 3 is the number of channels (red, green, and blue) for each spectrogram.

After many inspection methods for finding the optimized CNN-LSTM architecture, we configured the proposed model as shown in Table 2. A comparison between CNN, LSTM, and the proposed model to be persuasive, well-configured pure CNN, and pure LSTM models with a well-adjusted parameter setting were also designed.

The parameter settings of the proposed model are listed in Table 3. In this study, we considered a well-recommended optimizer and cost function.

TABLE 2. Configuration of the layers of the proposed model.

Layers	Size
Time Distributed Conv-2D	Feature maps = 16 Kernel size = 2×2 Stride =1×1
Time Distributed Conv-2D	Feature maps = 16 Kernel size = 2×2 Stride =1×1
Time Distributed Pooling	Max Pooling = 2×2
Time Distributed Flatten	16×496
LSTM hidden node	Hidden units = 500
Fully connected hidden node	Hidden units = 64
Dropout	0.2
Output node	Number of classes

TABLE 3. Parameter settings of the proposed model.

Parameters	Setting
Optimizer	Adam
Loss function	Cross entropy
Learning rate	0.001
Epoch	Dataset I: 500 Dataset II: 100
Batch size	Dataset I: 128

IV. DATASET SETUP

To ensure the generality of the proposed models, one private dataset was prepared and one public dataset was selected. The proposed models were applied separately to both datasets. The dataset description is as follows:

A. DATASET SETUP I

All keyword voices were collected from non-native English speakers' countries, namely, Ethiopia, Taiwan, and India. The

TABLE 4. Summary of the dataset description.

Training data terms	Description of the terms	Training data size
Original	The real voices excluding the augmented voices	Limited dataset-I: 5760 Very limited dataset-I: 2880 Dataset-II setup: 5000
Original + VC	The combination of original (real) voices and VC-generated voices	Limited dataset-I: 37,440 Very limited dataset-I: 34,560 Dataset-II setup: 50,000
Baseline augmentation	The generated mel-spectrograms data using affine transformation on original mel-spectrogram	Limited dataset-I: 57600 Very limited dataset-I: 28,800 Dataset-II setup: 60,000
VC + baseline augmentation	The mixture of the VC and affine transformations generated data	Limited dataset-I: 48,960 Very limited dataset-I: 40,320 Dataset-II setup: 60,000

total number of isolated keywords is 12 (“open”, “close”, “down”, “up”, “turn on”, “turn off”, “bed”, “bad”, “computer”, “hello”, “welcome”, and “university”). Most of these keywords were recorded from environments with background noise. The keyword voices were recorded at a sampling frequency of 16000 Hz, bit depth of 16 bits, and a monotype channel. The recording parameters of the datasets were fixed for all records. The recording time interval for the English keywords was between 1 and 1.5 seconds. After the recording, the audio files were stored as WAV files. All voices were recorded on laptop computers and Aver Media Microphone devices. All voices were collected from individuals with normal health status, and no person spoke emotionally during data collection. The voices of all but three speakers were recorded indoors. The total number of speakers was 8 Indian (5 females and 3 males), 10 Ethiopian (7 males and 3 females), and 12 Taiwanese (8 males and 4 females). Each keyword was spoken 20 times by all speakers. The dataset preparation method was purely speaker-independent. The 24 speakers were selected as limited training data and half of these limited training data (12 speakers) were assigned as very limited training data. The remaining six speakers were for test in both cases. The training and test data were collected separately, as illustrated in Fig. 4.

In our scenario, 12 speakers (four Taiwanese, four Ethiopian, and four Indian) and 10 speakers were selected from dataset setup-I and dataset setup-II for VC processing, respectively. A total of $12\times11\times12\times20 = 31{,}680$ and $10\times9\times10\times50 = 45$k new voices were generated for dataset setup-I and dataset setup-II, respectively, as shown in Fig. 5 and Table 4.

B. DATASET SETUP II

We also used the AudioMNIST dataset to evaluate the performance of the proposed model. Originally, the AudioMNIST consisted of 30000 audio recordings (9.5 hours) of spoken digits (0-9) in English, and each digit was spoken 50 times by 60 different speakers [53]. Since this study aims to

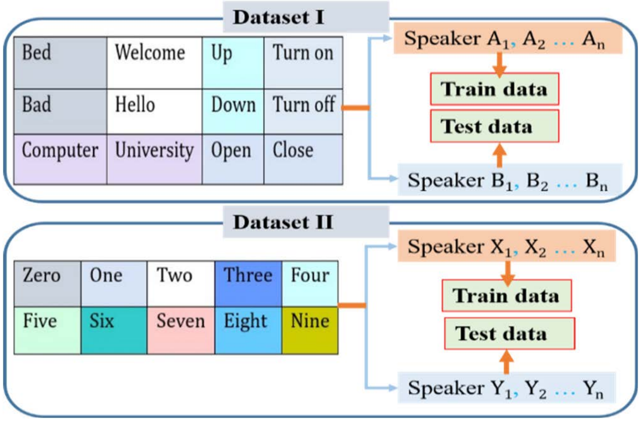

FIGURE 4. Speaker-independent keyword recognition dataset setups.

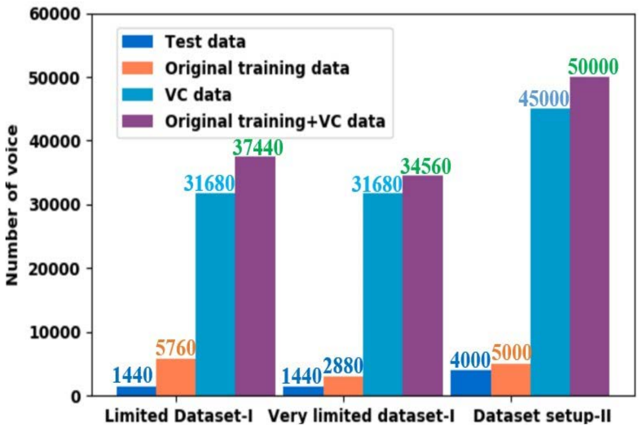

FIGURE 5. Dataset distribution.

improve speaker-independent keyword recognition on limited data, only 10 German speakers (6 male and 4 female) were assigned to the training data and 8 speakers (5 male and 3 female) were selected and assigned to the test data. All training data were selected from German native speakers, whereas test data were from German (three males and two females), South African (one male), Tamil (one female), and Arabic native speakers (one male).

Affine augmentation is a popular data augmentation technique for image recognition and spectrogram-based speech recognition using geometric transformations. We have used this very powerful augmentation technique as a baseline augmentation for evaluating the performance of the VC-based augmentation techniques. In this baseline augmentation, the affine transformation parameters are configured carefully for the comparison to be very convenient and unbiased. The 10 different mel-spectrogram images are generated for each of the original training samples during the baseline augmentation. For the proposed VC + baseline augmentation, 3 different mel-spectrogram images are generated from each real sample voice by applying baseline augmentation beside many voices which are generated using the VC methods.

V. RESULTS AND DISCUSSION

The experimental results were investigated using PyTorch for VC, Kera framework on the frontend, and TensorFlow framework as a backend for deep learning classification models using the Python programming language on the graphics processing unit (GPU). We used the NVIDIA GeForce RTX 2080 Ti GPU with 11 gigabytes (GB) of dedicated memory, where RTX stands for Ray Tracing Texel eXtreme and T is Titanium. Compute Unified Device Architecture (CUDA) Toolkit for the GPU-accelerated applications and NVIDIA CUDA deep neural network (cuDNN) GPU-accelerated libraries for deep neural networks were installed and configured on Windows 10 Intel 64-bit operating system. The original dataset setup-I is limited by itself, and half of this limited dataset is removed to further obtain a very limited dataset. We considered both limited and very limited as the baseline for comparing it to the proposed VC-based voice augmentation technique. Many voices were generated by the VC algorithm using the voices of a few speakers. The VC model was trained two times with the same model architecture for dataset-I and dataset-II separately. We assigned 2000 epochs for VC training phases to obtain a modified accent between the target and source speakers. A mean opinion score (MOS) subjective evaluation method [41] is selected for the naturalness and similarity evaluation of the converted voice. Five persons evaluated the naturalness and accent similarity between 25 converted sample voices and target voices. The evaluation score is 5 for excellent, 4 for good, 3 for fair, 2 for poor, and 1 for bad VC. Since we used the training data of parallel VC as test data again, the average MOS result is good for the naturalness and fair for the similarity. We examined the proposed model on two separate dataset setups to ensure that it performed well. In this study, the experiments for dataset-I were carried out for two cases, which are limited and very limited data.

TABLE 5. Model performance comparison on limited dataset-I.

Training data	CNN	LSTM	CNN-LSTM
Original	80	86	89
Original +VC	88	90	92
Baseline augmentation	88	90	91
VC + baseline augmentation	92	91	94

TABLE 6. Model performance comparison on very limited dataset-I.

Training data	CNN	LSTM	CNN-LSTM
Original	76	84	84
Original +VC	89	89	91
Baseline augmentation	85	86	88
VC + baseline augmentation	88	89	92

TABLE 7. Model performance comparison on dataset-II.

Training data	CNN	LSTM	CNN-LSTM
Original	92	86	93
Original +VC	94	94	94
Baseline augmentation	95	95	96
VC + baseline augmentation	96	94	97

In limited cases, all collected data (24 speakers) were taken as the original training data, and all 12 speakers' voices were converted to each other. The final dataset contained a mix of both original and converted voices. Performance comparison of the models and a summary of the results for limited data are presented in Table 5 and Fig. 6. The performance of the model is measured as follows:

$$Accuracy(\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{5}$$

where TP is the true positive, TN is the true negative, FP is the false positive, and FN is the false negative.

For very limited cases, we retained only half of the original training data (12 speakers). This is used to show how much VC is very useful for very limited data, as Table 6 and Fig. 7 show the result summary. The performance of the deep learning models was significantly improved by the proposed VC augmentation technique on very limited training data. Overall, the proposed model performed 94.2 % accuracy for keyword recognition on dataset setup-I. For dataset setup-II, all training voices (10 speakers) were selected for VC, and the results are shown in Table 7 and Fig. 6. The proposed VC augmentation method and CNN-LSTM model showed superior results on both dataset setups.

The deep learning models on a mix of original training data and converted data surprisingly improved the accuracy when compared to their performance on pure original data. For instance, the mix of 12 speakers' voices and their converted voices had better performance than the pure 24 speakers in the CNN model, as presented in Tables 6 and 7. Therefore, instead of collecting a large amount of data from many speakers, it is possible to compensate for this using

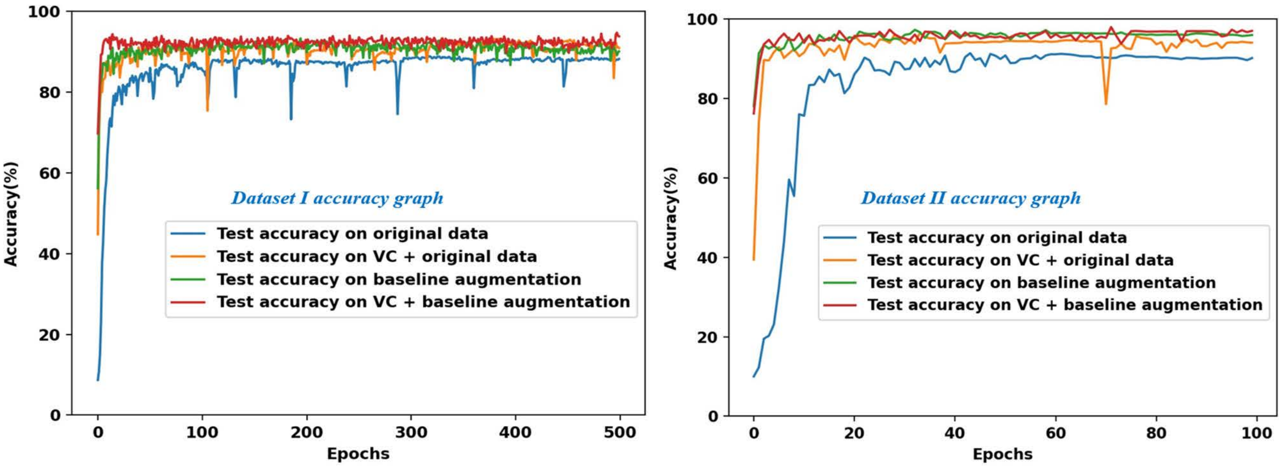

FIGURE 6. Proposed CNN-LSTM test accuracy graph.

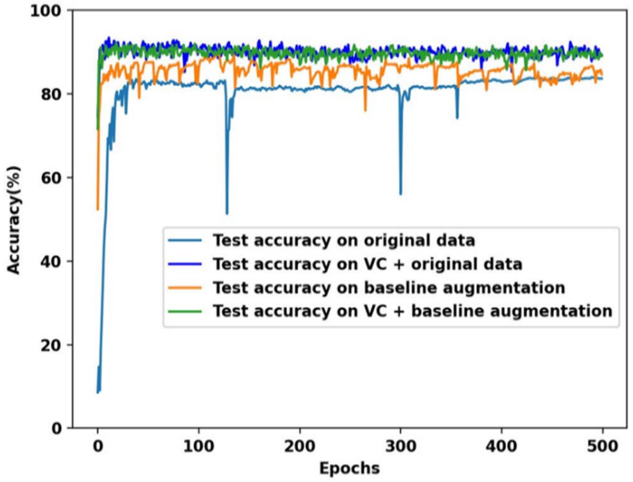

FIGURE 7. CNN-LSTM test accuracy graph on a very limited dataset I.

VC augmentation techniques. The overall results showed that the pure CNN and pure LSTM models were highly affected by the limited training dataset size when compared to our proposed CNN-LSTM model. Table 6 reported that the pure LSTM and the proposed models' performance are superior compared with the CNN model for very limited data without data augmentation techniques. Finally, we conclude that applying augmentation and regularization techniques to a mixture of both original and converted data enhanced the deep learning performance. We observe that the pure CNN performance has no stability for the testing data at different training iterations, whereas the proposed model has shown better consistency.

TABLE 8. Wilcoxon signed-rank test.

Pairwise Models	*p*-value	Comparison Evidence
CNN vs. LSTM	0.3394	Not-significant
CNN vs. CNN-LSTM	0.0032	Significant
LSTM vs. CNN-LSTM	0.0046	Significant
Baseline vs. Proposed augmentation	0.0117	Significant

TABLE 9. Nonparametric statistical comparisons among CNN, LSTM, AND CNN-LSTM.

Methods	*p*-value	Comparison Evidence
Friedman test	0.0001	Significant
Friedman Aligned test	0.0017	Significant
Quade test	0.0001	Significant

TABLE 10. Execution time comparison.

Dataset	Training Time (second)			Testing Time (second)		
	CNN	LSTM	CNN-LSTM	CNN	LSTM	CNN-LSTM
Limited Dataset–I	9521	6723	4585	0.33	0.52	0.32
Very Limited Dataset–I	6577	4626	3192	0.33	0.49	0.32
Dataset–II	2096	1512	1069	0.67	1.49	0.61

For good result analysis, we used the popular non-parametric statistical models' performance comparison techniques. The Wilcoxon signed-rank test is recommended for pairwise models' performance comparison, whereas the Friedman test, the Friedman aligned ranks test, and the Quade test for multiple algorithms [54], [55]. We chose the Wilcoxon signed ranks test for pairwise models' performance comparison, as Table 8 has reported. The Friedman test, the Friedman aligned ranks test, and the Quade test for three algorithms' performance comparison is depicted in Table 9. All these techniques were performed on the accuracy of all three different datasets. The significant level (p-value) < 0.05 indicates that one model is better than the other. All non-parametric statistical model performance comparisons' results showed that the proposed models were better than others.

The execution time analysis is very useful for trade-off the accuracy and computational time on the given models. The execution time of each model is shown in Table 10.

The execution time of the proposed algorithm (CNN-LSTM) is fast compared to pure CNN and pure LSTM models. The dataset-I (limited and very limited) and dataset-II training execution times are reported in 500 and 100 epochs, respectively.

VI. CONCLUSION

This paper presented a VC-based augmentation and CNN-LSTM model for robust speaker-independent keyword recognition. Many new modified versions of the original training voice were generated to increase the amount of training data. The experimental results showed that VC-based augmentation and the hybrid CNN-LSTM model improved speaker-independent keyword recognition. The mix of the original training data and converted data has comparable performance to affine transformation augmentation. The combination of affine transformations and VC augmentation has become more robust. The CNN-LSTM model has better accuracy and consistency than the pure CNN and pure LSTM. For very limited data without data augmentation techniques, the CNN model was highly affected compared with the pure LSTM and the proposed model. Extending this work for continuous speech recognition in a limited dataset size is under consideration for future work with some essential improvements to the current methodology.

REFERENCES

[1] S. Tabibian, ''A voice command detection system for aerospace applications,'' *Int. J. Speech Technol.*, vol. 20, no. 4, pp. 1049–1061, Dec. 2017, doi: 10.1007/s10772-017-9467-4.

[2] A. H. Michaely, X. Zhang, G. Simko, C. Parada, and P. Aleksic, ''Keyword spotting for Google assistant using contextual speech recognition,'' in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Dec. 2017, pp. 272–278, doi: 10.1109/ASRU.2017.8268946.

[3] P. M. Dias and K. Jayakody, ''Virtual assistant in native language,'' in *Proc. IEEE Asia–Pacific Conf. Geosci., Electron. Remote Sens. Technol. (AGERS)*, Dec. 2020, pp. 16–18, doi: 10.1109/AGERS51788.2020.9452751.

[4] X. Lv, M. Zhang, and H. Li, ''Robot control based on voice command,'' in *Proc. IEEE Int. Conf. Autom. Logistics*, Sep. 2008, pp. 2490–2494, doi: 10.1109/ICAL.2008.4636587.

[5] T. Q. Nguyen, P. Nauth, and S. Sharan, ''Control of autonomous mobile robot using voice command,'' in *Proc. ARW OAGM Workshop*, Jan. 2019, pp. 1–3, doi: 10.3217/978-3-85125-663-5-24.

[6] H. Lee, S. Chang, D. Yook, and Y. Kim, ''A voice trigger system using keyword and speaker recognition for mobile devices,'' *IEEE Trans. Consum. Electron.*, vol. 55, no. 4, pp. 2377–2384, Nov. 2009, doi: 10.1109/TCE.2009.5373813.

[7] M. Sidiq, W. T. A. Budi, and S. Sa'adah, ''Vomma: Android application launcher using voice command,'' in *Proc. 3rd Int. Conf. Inf. Commun. Technol. (ICoICT)*, May 2015, pp. 49–53, doi: 10.1109/ICoICT.2015.7231395.

[8] R. D. H. Arifin and R. Sarno, ''Door automation system based on speech command and PIN using Android smartphone,'' in *Proc. Int. Conf. Inf. Commun. Technol. (ICOIACT)*, Mar. 2018, pp. 667–672, doi: 10.1109/ICOIACT.2018.8350715.

[9] Z. Kons, S. Shechtman, A. Sorin, R. Hoory, C. Rabinovitz, and E. D. S. Morais, ''Neural TTS voice conversion,'' in *Proc. IEEE Spoken Lang. Technol. Workshop (SLT)*, Dec. 2018, pp. 290–296, doi: 10.1109/SLT.2018.8639550.

[10] A. Kain and M. W. MacOn, ''Spectral voice conversion for text-to-speech synthesis,'' in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, vol. 1, no. 3, May 1998, pp. 285–288, doi: 10.1109/ICASSP.1998.674423.

[11] O. Turk, O. Buyuk, A. Haznedaroglu, and L. M. Arslan, ''Application of voice conversion for cross-language rap singing transformation,'' in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 3597–3600.

[12] F. Villavicencio and J. Bonada, ''Applying voice conversion to concatenative singing-voice synthesis,'' in *Proc. Interspeech*, Sep. 2010, pp. 2162–2165, doi: 10.21437/interspeech.2010-596.

[13] D. Felps, H. Bortfeld, and R. Gutierrez-Osuna, ''Foreign accent conversion in computer assisted pronunciation training,'' *Speech Commun.*, vol. 51, no. 10, pp. 920–932, 2009.

[14] Z. Luo, J. Chen, T. Takiguchi, and Y. Ariki, ''Emotional voice conversion using neural networks with arbitrary scales f0 based on wavelet transform,'' *EURASIP J. Audio, Speech, Music Process.*, vol. 2017, no. 1, pp. 1–13, Dec. 2017, doi: 10.1186/s13636-017-0116-2.

[15] K. Zhou, B. Sisman, M. Zhang, and H. Li, ''Converting anyone's emotion: Towards speaker-independent emotional voice conversion,'' in *Proc. Interspeech*, Oct. 2020, pp. 3416–3420.

[16] Y.-J. Chan, C.-J. Peng, S.-S. Wang, H.-M. Wang, Y. Tsao, and T.-S. Chi, ''Speech enhancement-assisted StarGAN voice conversion in noisy environments,'' 2021, *arXiv:2110.09923*.

[17] H. Doi, T. Toda, K. Nakamura, H. Saruwatari, and K. Shikano, ''Alaryngeal speech enhancement based on one-to-many eigenvoice conversion,'' *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 1, pp. 172–183, Jan. 2014, doi: 10.1109/TASLP.2013.2286917.

[18] T. Toda, L.-H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, and J. Yamagishi, ''The voice conversion challenge 2016,'' in *Proc. Interspeech*, Sep. 2016, pp. 1632–1636, doi: 10.21437/Interspeech.2016-1066.

[19] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, and Z. Ling, ''The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods,'' in *Proc. Speaker Lang. Recognit. Workshop (Odyssey)*, Jun. 2018, pp. 195–202, doi: 10.21437/odyssey.2018-28.

[20] Z. Yi, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z.-H. Ling, and T. Toda, ''Voice conversion challenge 2020—Intra-lingual semi-parallel and cross-lingual voice conversion—,'' in *Proc. Joint Workshop Blizzard Challenge Voice Convers. Challenge*, Oct. 2020, pp. 80–98, doi: 10.21437/vcc_bc.2020-14.

[21] G. Jin, M. T. Johnson, J. Liu, and X. Lin, ''Voice conversion based on Gaussian mixture modules with minimum distance spectral mapping,'' in *Proc. 5th Int. Conf. Inf. Sci. Technol. (ICIST)*, Apr. 2015, pp. 356–359, doi: 10.1109/ICIST.2015.7288996.

[22] T. Toda, A. W. Black, and K. Tokuda, ''Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory,'' *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 8, pp. 2222–2235, Nov. 2007, doi: 10.1109/TASL.2007.907344.

[23] S. Desai, A. W. Black, B. Yegnanarayana, and S. Member, ''Networks for voice conversion,'' *Language (Baltim)*, vol. 18, no. 5, pp. 954–964, 2010.

[24] B. Sisman, M. Zhang, M. Dong, and H. Li, ''On the study of generative adversarial networks for cross-lingual voice conversion,'' in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Dec. 2019, pp. 144–151, doi: 10.1109/ASRU46091.2019.9003939.

[25] T. Kaneko and H. Kameoka, ''CycleGAN-VC: Non-parallel voice conversion using cycle-consistent adversarial networks,'' in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 2100–2104.

[26] M. Zhang, B. Sisman, L. Zhao, and H. Li, ''DeepConversion: Voice conversion with limited parallel training data,'' *Speech Commun.*, vol. 122, pp. 31–43, Sep. 2020, doi: 10.1016/j.specom.2020.05.004.

[27] B. Chen, Z. Xu, and K. Yu, ''Data augmentation based non-parallel voice conversion with frame-level speaker disentangler,'' *Speech Commun.*, vol. 136, pp. 14–22, Jan. 2022, doi: 10.1016/j.specom.2021.10.001.

[28] T. Ishihara and D. Saito, ''Attention-based speaker embeddings for one-shot voice conversion,'' in *Proc. Interspeech*, Oct. 2020, pp. 806–810, doi: 10.21437/Interspeech.2020-2512.

[29] G. Van Houdt, C. Mosquera, and G. Napoles, ''A review on the long short-term memory model,'' in *Artificial Intelligence Review*, vol. 53. Berlin, Germany: Springer-Verlag, May 2020, pp. 5929–5955, doi: 10.1007/s10462-020-09838-1.

[30] X. Li and Z. Zhou. (2019). *Speech Command Recognition With Convolutional Neural Network*. Stanford CS 229 Projects. [Online]. Available: http://cs229.stanford.edu/proj2017/final-reports/5244201.pdf

[31] D. M. Waqar, T. S. Gunawan, M. Kartiwi, and R. Ahmad, ''Real-time voice-controlled game interaction using convolutional neural networks,'' in *Proc. IEEE 7th Int. Conf. Smart Instrum., Meas. Appl. (ICSIMA)*, Aug. 2021, pp. 76–81, doi: 10.1109/ICSIMA50015.2021.9526318.

[32] Y. A. Wubet and K.-Y. Lian, "A hybrid model of CNN-SVM for speakers' gender and accent recognition using english keywords," in *Proc. IEEE Int. Conf. Consum. Electron.-Taiwan (ICCE-TW)*, Sep. 2021, pp. 1–2, doi: 10.1109/ICCE-TW52618.2021.9603210.

[33] A. N. Cayir and T. S. Navruz, "Effect of dataset size on deep learning in voice recognition," in *Proc. 3rd Int. Congr. Hum.-Comput. Interact., Optim. Robotic Appl. (HORA)*, Jun. 2021, pp. 1–5.

[34] X. Yang, H. Yu, and L. Jia, "Speech recognition of command words based on convolutional neural network," in *Proc. Int. Conf. Comput. Inf. Big Data Appl. (CIBDA)*, Apr. 2020, pp. 465–469.

[35] J. Louis K. E. Fendji, D. C. M. Tala, B. O. Yenke, and M. Atemkeng, "Automatic speech recognition and limited vocabulary: A survey," 2021, *arXiv:2108.10254*.

[36] S. Shahnawazuddin, N. Adiga, K. Kumar, A. Poddar, and W. Ahmad, "Voice conversion based data augmentation to improve children's speech recognition in limited data scenario," in *Proc. Interspeech*, Oct. 2020, pp. 4382–4386, doi: 10.21437/Interspeech.2020-1112.

[37] D. K. Singh, P. P. Amin, H. B. Sailor, and H. A. Patil, "Data augmentation using CycleGAN for end-to-end children ASR," in *Proc. 29th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2021, pp. 511–515, doi: 10.23919/EUSIPCO54536.2021.9616228.

[38] M. Baas and H. Kamper, "Voice conversion can improve ASR in very low-resource settings," 2021, *arXiv:2111.02674*.

[39] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech Signal Process.*, vol. ASSP-33, no. 2, pp. 443–445, Apr. 1985, doi: 10.1109/TASSP.1985.1164550.

[40] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[41] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "ACVAE-VC: Non-parallel voice conversion with auxiliary classifier variational autoencoder," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 27, no. 9, pp. 1432–1443, Sep. 2019, doi: 10.1109/TASLP.2019.2917232.

[42] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Trans. Inf. Syst.*, vol. E99.D, no. 7, pp. 1877–1884, 2016, doi: 10.1587/TRANSINF.2015EDP7457.

[43] J. Qin, W. Pan, X. Xiang, Y. Tan, and G. Hou, "A biological image classification method based on improved CNN," *Ecolog. Informat.*, vol. 58, Jul. 2020, Art. no. 101093, doi: 10.1016/j.ecoinf.2020.101093.

[44] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, Oct. 2017.

[45] M. Dawodi, J. A. Baktash, T. Wada, N. Alam, and M. Z. Joya, "Dari speech classification using deep convolutional neural network," in *Proc. IEEE Int. IoT, Electron. Mechatronics Conf. (IEMTRONICS)*, Sep. 2020, pp. 2020–2023.

[46] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 131–135.

[47] Y. Liu, Z. Su, H. Li, and Y. Zhang, "An LSTM based classification method for time series trend forecasting," in *Proc. 14th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, Jun. 2019, pp. 402–406, doi: 10.1109/ICIEA.2019.8833725.

[48] J. H. Wang, T. W. Liu, X. Luo, and L. Wang, "An LSTM approach to short text sentiment classification with word embeddings," in *Proc. 30th Conf. Comput. Linguist. Speech Process. (ROCLING)*, Oct. 2018, pp. 214–223.

[49] A. Graves, N. Jaitly, and A.-R. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand.*, Dec. 2013, pp. 273–278, doi: 10.1109/ASRU.2013.6707742.

[50] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid CNN-LSTM model for short-term individual household load forecasting," *IEEE Access*, vol. 8, pp. 180544–180557, 2020, doi: 10.1109/ACCESS.2020.3028281.

[51] J. Zhu, H. Chen, and W. Ye, "A hybrid CNN–LSTM network for the classification of human activities based on micro-Doppler radar," *IEEE Access*, vol. 8, pp. 24713–24720, 2020, doi: 10.1109/ACCESS.2020.2971064.

[52] A. Ankita, S. Rani, A. K. Bashir, A. Alhudhaif, D. Koundal, and E. S. Gunduz, "An efficient CNN-LSTM model for sentiment detection in #BlackLivesMatter," *Expert Syst. Appl.*, vol. 193, pp. 1–8, Jan. 2022, doi: 10.1016/j.eswa.2021.116256.

[53] S. Becker, M. Ackermann, S. Lapuschkin, K.-R. Müller, and W. Samek, "Interpreting and explaining deep neural networks for classification of audio signals," 2018, *arXiv:1807.03418*.

[54] K. Stapor, "Evaluation of classifiers: Current methods and future research directions," in *Proc. Ann. Comput. Sci. Inf. Syst.*, Sep. 2017, pp. 37–40, doi: 10.15439/2017f530.

[55] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Dec. 2006.

YESHANEW ALE WUBET received the B.S. degree in computer science from Assosa University, Assosa, Ethiopia, in 2015, and the M.S. degree in electrical engineering and computer science from the National Taipei University of Technology, Taipei, Taiwan, in 2019, where he is currently pursuing the Ph.D. degree in electrical engineering and computer science.

His research interests include speech recognition, image processing, embedded computing, and machine learning.

KUANG-YOW LIAN (Member, IEEE) received the B.S. degree in engineering science from the National Cheng Kung University, Tainan, Taiwan, in 1984, and the Ph.D. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, in 1993.

From 1994 to 2007, he was an Associate Professor, and the Chair of the Department of Electrical Engineering, Chung Yuan Christian University, Zhongli, Taiwan. He is currently a Distinguished Professor with the Department of Electrical Engineering, National Taipei University of Technology. He also served as the Chair, from 2009 to 2012. His research interests include smart sensor technology, smart living devices, machine learning, robotics, and control system applications.

Prof. Lian's awards and honors include the Chinese Automatic Control Society (CACS) Fellow, in 2015, the CACS Outstanding Automatic Control Engineering Award, in 2012, the 2014 and 2017 Macronix Gold Silicon Best Advisor, and the Future Science and Technology Award of the Annual Event in Taiwan's Scientific Research, in 2021.