Neurocomputing 418 (2020) 211–220

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

SpeakerGAN: Speaker identification with conditional generative adversarial network

Liyang Chen [a,b], Yifeng Liu [c], Wendong Xiao [a,b,*], Yingxue Wang [c,d,*], Haiyong Xie [d,c,e]

[a] *School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China*
[b] *Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, Beijing 100083, China*
[c] *National Engineering Laboratory for Public Safety Risk Perception and Control by Big Data (NEL-PSRPC), Beijing 100041, China*
[d] *Advanced Innovation Center for Human Brain Protection, Capital Medical University, Beijing 100054, China*
[e] *University of Science and Technology of China, Hefei, Anhui 230027, China*

ARTICLE INFO

Article history:
Received 17 January 2020
Revised 5 August 2020
Accepted 21 August 2020
Available online 1 September 2020
Communicated by Steven Hoi

Keywords:
Speaker identification
Generative adversarial networks
Deep neural networks

ABSTRACT

Current methods based on the traditional i-vectors and deep neural network (DNN) have shown effectiveness on the speaker identification task, especially with the corpus of large scale. However, when the size of the training dataset is small, the overfitting problem may happen and lead to performance degradation. Besides, the robust identification still remains a challenging problem even under the less strict requirements. This paper proposes a novel approach, SpeakerGAN, for speaker identification with the conditional generative adversarial network (CGAN). It allows the adversarial networks for distinguishing real/fake samples and predicting class labels simultaneously. We configure the generator and the discriminator in SpeakerGAN with the gated convolutional neural network (CNN) and the modified residual network (ResNet) to obtain generated samples of high diversity as well as increase the network capacity. The multiple loss functions are combined and optimized to encourage the correct mapping and accelerate the convergence. Experimental results show that SpeakerGAN reduces the classification error rate by 87% and 16% compared with the traditional i-vector system and the state-of-the-art DNN based method. Under the scenario of limited training data, SpeakerGAN obtains significant improvement over the baselines. In the case of taking 1.6 s of each speaker for testing, SpeakerGAN achieves the identification accuracy of 98.20%, which suggests the promise for short-utterance speaker identification.

1. Introduction

Speaker identification (SI), also known as the voice-print recognition is a technology to identify a speaker by his or her speech. An utterance from a speaker conveys information of hundreds of words, but only corresponds to a constant identity. This technique has gained a lot of attention and been applied in various fields, such as voice conversion, speaker diarization, biometrics authentication, crime forensics, financial security and mobile payment.

SI can be divided into text-dependent and text-independent SI tasks according to the limitation of the speech dictionary. The text-dependent SI systems require the speech to be produced from a fixed text phrase. The text-independent SI systems utilize no prior knowledge of the speech to be spoken. Our work concentrates on the text-independent SI. It is more difficult and challenging but exhibits to be more meaningful in real application scenarios.

From statistical modeling to neural networks, research and development on SI have been undertaken for decades. Non-parametric models, including template matching and vector quantization [1], were first introduced to deal with SI. However, they cannot meet the practical requirements of industrial application, because the variables or vectors that represent speakers are too sensitive to changes of channel and background noise. With large corpora publicly available, scholars began to use parametric frameworks for acoustic modeling. Hidden Markov Model (HMM) [2] was employed to assume the context and short-term sequence information in speech as a Markov process with unobservable (i.e., hidden) states. To better approximate the distribution of speaker characteristics, Gaussian mixture model (GMM) [3] was applied in SI. Many subsequent researches have been investigated and expanded based on GMM. Reynolds proposed the universal background model (UBM) [4], establishing the GMM-UBM system,

[*] Corresponding authors at: School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China (W. Xiao). National Engineering Laboratory for Public Safety Risk Perception and Control by Big Data (NEL-PSRPC), Beijing 100041, China (Y. Wang).

E-mail addresses: wdxiao@ustb.edu.cn (W. Xiao), wangyingxue@csdslab.net (Y. Wang).

https://doi.org/10.1016/j.neucom.2020.08.040
0925-2312/© 2020 Elsevier B.V. All rights reserved.

to alleviate the data sparsity problem. Support vector machines (SVM) [5,6] and factor analysis (FA) [7] were studied to perform channel compensation and dimensionality reduction. In 2010, Dehak et al. [8] defined a new low-dimensional space that models both speaker and channel variabilities. In this space, a given speech utterance is represented by a new vector named identity vector (i-vector). The i-vector framework improves the robustness and generalization ability of SI system, and has become the dominant approach for modeling in text-independent SI applications over several years.

With the rapid development of the deep neural network (DNN), many DNN based methods have shown superior performance over the traditional approaches. In Ref. [9], Lei et al. proposed a framework which produces frame posteriors for the computation of i-vectors using a DNN instead of GMM-UBM. Inspired by the i-vector system, d-vectors [10] are extracted as speaker specific features from the last hidden layer of a DNN, which has been trained to classify speakers at the frame level. Similar attempts [11–14] which handle variable length inputs at the level of segment or utterance have also been brought to improve the paradigm. The x-vector in [13,14] is proposed as a strong contender for the speaker representation and is considered to supplant the i-vector system by many researchers.

Due to the remarkable achievement in image processing and speech recognition, convolutional neural network (CNN) [15–20] is introduced to capture correlations in time and frequency domain of acoustic features. In Refs. [11,21,22], long short-term memory (LSTM) and recurrent neural network (RNN) are utilized to focus more on the long-term temporal dependencies across speech. As a mechanism to emphasize the most relevant elements of the input sequence, many studies adopt attention layers [20,22] to improve the SI system performance.

On the whole, deeper networks and more complicated architectures always lead to the identification enhancement. However, these approaches mentioned above appear to be in lack of scalability with the scale of the training corpus, and there is still a margin for improvement. Either the conventional i-vector system or the state-of-the-art DNN is aimed to learn a mapping from the speaker utterance to the speaker representations or speaker identities. In the absence of sufficient training data, these data-driven modeling methods are usually confronted with rapid performance degradation. Moreover, the problem of overfitting on the limited corpus reduces the generalization ability of the model.

Motivated by these facts, this paper proposes a novel SI approach, called SpeakerGAN, using a modified conditional generative adversarial network (CGAN). Compared with the traditional CGAN which is designed only for producing samples, we adapt the CGAN for simultaneously learning a generative model and a classifier. The generator in SpeakerGAN still learns the target data distribution similar to the regular CGAN. The discriminator takes the generated samples and real acoustic features as inputs, and outputs the class labels of real samples in addition to the real/fake probability. To provide more indistinguishable generation for the discriminator, aside from the adversarial loss, the Huber loss is involved to encourage explicit density estimation. We configure the generator and discriminator of the advanced architectures using gated CNNs and the modified residual network (ResNet) to allow for sufficient network capacity as well as faster convergence.

We evaluate our approach on the dataset of Librispeech-100 for the text-independent SI task. The baselines include i-vector, x-vector, CNN and LSTM. Experimental results show that the proposed method outperforms the state-of-the-art SI systems with sufficient data provided. The necessity of modules in the SpeakerGAN pipeline is confirmed by removing them separately. In the case of limited training data, the SpeakerGAN achieves greater advantage over the baselines. We also investigate the identification performance when different front-end features of different dimensions are used as the speaker representations. Interestingly, the SpeakerGAN also demonstrates great potential in short-utterance based SI, due to the preprocessing of speech utterances and the reasonable design of network architectures.

The paper is organized as follows. In Section 2, related works about classical SI methods and GAN based studies are described. In Section 3, we review the basic GAN and CGAN. Section 4 demonstrates the proposed SpeakerGAN and the architecture of the generator and discriminator in SpeakerGAN. Experiments are carried out in Section 5 and the results are presented in Section 6. Section 7 summarizes the paper and illustrates the future work.

2. Related works

As the traditional baseline, the i-vector system still holds the basic position and provides an effective heuristics pattern for the SI task. In the sense of converting high dimensional statistics to low dimensional vectors by independently training on a universal dataset, the i-vector based SI models have already alleviated the dependence on large amounts of data as well as shown effective identification improvement. Prior study found that the embeddings [10] leverage large-scale training datasets better than i-vectors. Snyder et al. [13,14] proposed x-vector by adding the statistics pooling layer and forming a fixed-length representation of the input utterance at the segment level. This model built on the time-delay neural network (TDNN) has gained superior performance over the traditional i-vector system.

There have been deep learning (DL) based efforts as well. Deep Speaker [16], which draws network architectures from speech recognition systems, shows significant efficacy on the text-independent SI and transfers well across spoken languages. In Refs. [15,17], spectrograms that extracted from raw audios after fast Fourier transform (FFT) without applying mel-scale filter bank, are directly fed into CNNs. This model outperforms other methods on the dataset collected in the wild. Ref. [23] proposes the SincNet that encourages the first convolutional layer to discover more meaningful filters based on parameterized sinc functions. Speaker characteristics like pitch and formants, are precisely extracted through the designed SincNet filters.

Recently, GANs have been studied vigorously and applied in many speech or audio related tasks, which mainly focus on domain transformation and data generation, such as speech enhancement [24], voice conversion [25–27] and voice synthesis [28]. The theoretical premises and algorithms of GANs to be extended for classification have already been discussed in Refs. [29–31] on image classification. In Ref. [30], Springenberg proposed the categorical GAN (CatGAN) for learning a discriminative classifier from unlabeled or partially labeled data. Odena [31] extended GANs to the semi-supervised context by forcing the discriminator network to output class labels. In Ref. [29], Salimans et al. further illustrated the continuation and refinement of previous effort on semi-supervised learning. Recent researches on GANs for the spoken language identification (LID) task have shown promising results [32,33], where the CGANs are used to improve regularization of the models. Instead of only inputting noise to the generator, the CGAN in Ref. [32] uses the same layer to predict both the class label while [33] creates separate output layers according to separate loss functions.

Similar attempts have been made for the speaker recognition task. In Ref. [34], due to the mismatch between the distributions of training and test data, a domain adaptation technique using GAN was proposed to compensate for domain or covariate shift. The unsupervised domain adaptation technique is also investigated in Ref. [35] with a cycle-consistent GAN (CycleGAN), when a limited amount of target domain data is available. To improve the performance of the i-vector system with short utterances, an i-vector compensation method using GAN is proposed in Ref. [36], where the generator tries to produce a reliable i-vector from an unreliable one. In Ref. [37], the CGAN is applied in speech enhancement and presents a substantial improvement in the evaluation of speaker verification. To extract embeddings more related to the speaker identities, Ding and He [38] proposed an enhanced triplet method by combining the multitasking learning and GAN.

Inspired by the previous works about GAN theories in Ref. [29], this paper establishes SpeakerGAN referred from Refs. [32,34]. To the best our knowledge, there have been no methods that directly utilize CGAN for the SI task.

3. Generative adversarial networks

3.1. Basic GAN

Generative adversarial network (GAN) was proposed by Goodfellow et al. [39] in 2014, which is a framework for training generative models via an adversarial process. The GAN consists of two models: a generator G that learns a mapping from random noise variables z (follows uniform or Gaussian distribution $p_z(z)$) to the data space $G(z)$, and a discriminator D that distinguishes whether the inputs come from real data distribution $p_{data}(x)$ or fake data $G(z)$. The objective of GAN can be expressed as

$$\begin{aligned}\min_G \max_D V(G,D) &= \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] \\ &\quad + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))],\end{aligned} \tag{1}$$

where $D(x)$ denotes the probability that inputs come from the real data. In the process of jointly optimizing G and D, the generator tries to minimize this objective, while the discriminator tries to maximize it, like a minimax two-player game.

3.2. Conditional GAN

The CGAN is a variant of GAN, which aims to let the generator G produce $G(c,z)$ from the condition c and random noise z. The discriminator D in CGAN needs to classify (x,c) as the real and $(G(z,c),c)$ as the fake, where x corresponds to real samples from $p_{data}(x)$. The objective is written as

$$\begin{aligned}\min_G \max_D V(G,D) &= \mathbb{E}_{x\sim p_{data}(x)}[\log D(x,c)] \\ &\quad + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z,c),c))],\end{aligned} \tag{2}$$

where the condition c could denote class labels or other data modalities and it is introduced to direct the generation process, producing certain results as well as significantly improving the quality of outputs. In this paper, the real samples x are directly utilized as the condition c and the random noise z is abandoned for showing no effectiveness in the experiments. The generated samples $G(x)$ and real samples x are fed into the discriminator separately, which is different from the regular CGAN.

4. SpeakerGAN for speaker identification

4.1. Basic principle of SpeakerGAN

The basic GAN and CGAN both focus on generating excellent output samples and the discriminator is only designed to predict whether the inputs are real or fake. Since GANs achieved great success in generation, it is a good idea to make GANs solve classification tasks. When there is no sufficient data for training, it seems reasonable to use generated samples from $p_g(x)$, which is close to the real data distribution, to improve generalization performance for the model. In this work, we investigate the CGAN as a classifier by enabling the network to learn on additional unlabeled examples. Fig. 1 illustrates the framework of the proposed SpeakerGAN. The discriminator in SpeakerGAN has a "real/fake" output. The standard classifier part outputs a N-dimensional vector $l = \{l_1, \ldots, l_N\}$, where l_k is the probability $p_{model}(y=k|x)$ that the input x belongs to class k. The SpeakerGAN combines the discriminator and classifier by letting the classifier take samples from the generator as inputs and have $N+1$ output units, where l_{N+1} corresponds to the probability $p_{model}(y=N+1|x)$ that the inputs are real. Thus, the loss function is split into a classification loss $\mathcal{L}_{class}$ and an adversarial loss $\mathcal{L}_{adv}$

$$\mathcal{L}_{class}(D) = -\mathbb{E}_{x,y\sim p_{data}(x,y)} \sum_{k=1}^{N} y_k \log l_k, \tag{3}$$

$$\begin{aligned}\mathcal{L}_{adv}(G,D) = &-\mathbb{E}_{x\sim p_g(x)}[\log p_{model}(y=k+1|x)] \\ &-\mathbb{E}_{x\sim p_{data}(x)}[\log[1-p_{model}(y=k+1|x)]].\end{aligned} \tag{4}$$

$\mathcal{L}_{class}$ denotes the cross-entropy loss using all labeled data, where y is the label vector of a training sample. The generated samples are given the labels, which belong to the original real samples they learned from. $\mathcal{L}_{adv}$ is the adversarial loss when learning the target distribution. To stabilize the training and overcome the problem of vanishing gradients, the least square GAN (LSGAN) [40] is adopted. $\mathcal{L}_{adv}$ is then split into the losses for the discriminator and generator

$$\mathcal{L}_{adv}(D) = \mathbb{E}_{x\sim p_{data}(x)}[(D(x)-1)^2]\mathbb{E}_{x\sim p_{data}(x)}[D(G(x))^2]. \tag{5}$$

$$\mathcal{L}_{adv}(G) = \mathbb{E}_{x\sim p_{data}(x)}[(D(G(x))-1)^2]. \tag{6}$$

In the previous study [40], it has shown that the quality of the regular GAN outputs varies greatly. Since the adversarial loss does not help preserve the contextual information of real inputs, a defective data distribution might be learned when there are many mappings that can satisfy GAN. To guarantee the generation to be near the ground-truth output, the Huber loss is introduced. The Huber loss is a loss function used in robust regression that is less sensitive to outliers in data than the squared error loss. In recent research [41], the Huber loss was proved to generate images of higher resolution. Given an input of size $H \times W$ and a corresponding fake input $G(x)$ of the same size, the Huber loss is defined as

$$\mathcal{L}_{Huber}(G) = \frac{1}{H \cdot W} \sum_{i=1}^{H} \sum_{j=1}^{W} \delta(i,j), \tag{7}$$

$$\delta(i,j) = \begin{cases} \frac{1}{2}[x_{ij} - G(x)_{ij}]^2, & x_{ij} - G(x)_{ij} \leqslant 1 \\ |x_{ij} - G(x)_{ij}|, & otherwise, \end{cases} \tag{8}$$

where $x_{ij} - G(x)_{ij}$ is named as residual and the threshold of residual can be changed.

The overall objectives to be minimized for the generator and discriminator is respectively written as

$$\mathcal{L}(G) = \mathcal{L}_{adv}(G) + \mathcal{L}_{Huber}(G), \tag{9}$$

$$\mathcal{L}(D) = \lambda_1 \mathcal{L}_{adv}(D) + \lambda_2 \mathcal{L}_{class}(D), \tag{10}$$

where λ_1 and λ_2 are the trade-off parameters that encourage unlabeled learning and classification, separately.

4.2. Network architecture of SpeakerGAN

We adopt different architectures for the generator and discriminator in SpeakerGAN. The generator is always designed like an encoder-decoder structure as illustrated in Fig. 1. In encoding, the generator is expected to capture both the temporal and spatial features, while it is supposed to produce feature sequences of high

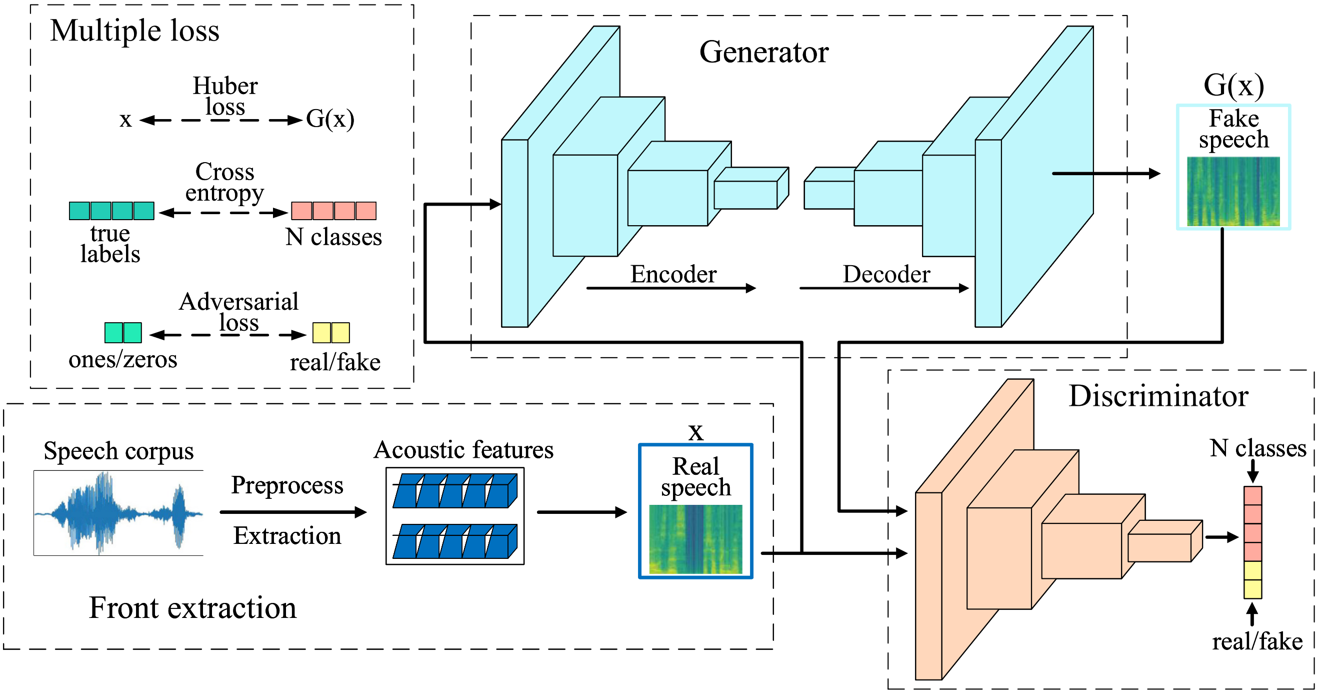


Fig. 1. Framework of SpeakerGAN. The front extraction part extracts FBanks from the real speech samples. The generator takes the real FBanks x as inputs and produces fake samples $G(x)$. The discriminator is then fed with the real and generated FBanks to predict class labels and distinguish between the real and fake samples. The adversarial loss in Multiple loss actually denotes the formulation of LSGAN [40]. The dashed lines with arrows denote calculating loss between two objects, and the solid lines denote the flow of information.

quality when decoding. The discriminator requires to model the sequential features of speakers as well as have enough depth to process training data of large scale. Raw speech is preprocessed with the operation described in Section 5.1. The speaker representations as feature sequences are then obtained. The generator takes real sequences as the condition for inputs, and produces the fake samples of the same size after passing through a series of convolutional and shuffler layers that progressively downsample and upsample. The discriminator takes the generated samples and real acoustic features from the corpus as inputs, and outputs the discrimination of real/fake along with the N classes.

4.2.1. Generator design

The generator is developed to extract relationships from inputs and reconstruct feature sequences of speakers. In some GANs applied speech tasks, e.g., voice conversion [42] and language identification [32], the generators are usually constructed with simple convolutional layers or fully connected layers. These generators only capture relationships among feature dimension and the generated samples are in lack of consistency. An effective way to solve this problem would be to introduce the RNN, but it is time-consuming due to the difficulty of parallel computing. For these reasons, we configure the generator using gated CNNs [43], which allow for both sequential structure and faster convergence. This idea was previously explored in [25], and achieved competitive performance.

The block of the gated CNN wraps the convolution and the gated linear unit (GLU) instead of the rectified linear unit (ReLU), as shown in Fig. 2. GLU is an activation function and the hidden layer output $h_l(x)$ can be calculated as

$$h_l(x) = (x * W + b) \otimes \sigma(x * V + c), \tag{11}$$

where x is the input of layer h_l, W, b, V and c are the parameters of linear projection layers (e.g., convolutional layers), σ denotes the sigmoid function and $\otimes$ is the element-wise product between matrices. With this gating mechanism, information passing in the hierarchy can be controlled depending on the previous layer states. At the same time, GLUs are capable of featuring long-term dependencies, similar to LSTM. The encoder part of the generator consists of all gated CNN blocks with different convolutional kernel sizes and strides.

After extracting patterns from inputs, upsampling layers composed of pixel shuffler [44] are used to increase the feature map dimension. In the field of computer vision processing, pixel shuffler is effective for high-resolution image reconstruction. Table 1 shows the details of the generator architecture.

4.2.2. Discriminator design

As for the discriminator, we prefer deeper networks to classify speakers. However, training deep neural networks is computationally expensive and difficult. This paper modifies the ResNet [45] to accelerate the training. ResNets have been applied in many SI systems [16,17] and are known for good classification performance on

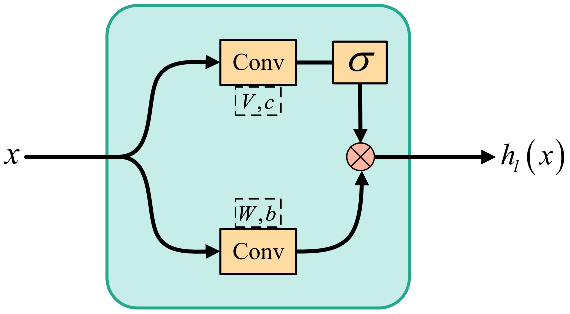


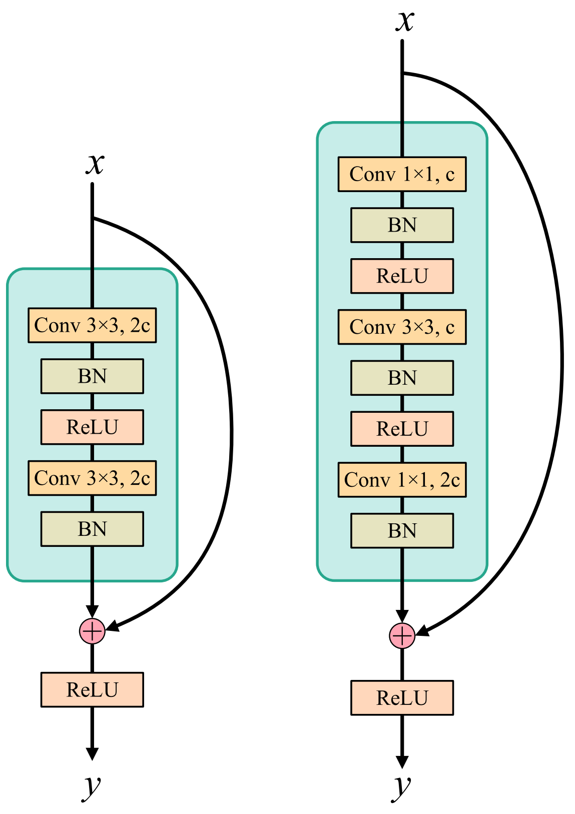
Fig. 2. The architecture of the gated CNN block. The sigmoid function acts as a gate to control the flow of the input x. The output is computed as the element-wise product between the gated and convoluted matrices.

Table 1
Generator architecture. The generator comprises two components: downsampling and upsampling layers. The first convolutional layer is used to discover more features in both time and frequency domains.

Layer name	Structure	Stride	Output dim
conv1	15 × 1, 256	1 × 1	(160,256)
gate1	–	–	(160,256)
Downsample1			
conv2	5 × 1, 512	2 × 1	(80,512)
gate2	–	–	(80,512)
Downsample2			
conv3	5 × 1, 1024	2 × 1	(40,1024)
gate3	–	–	(40,1024)
Upsample1			
conv4	5 × 1, 1024	1 × 1	(40,1024)
shuffler1	5 × 1, 1024	2 × 1	(80,512)
gate4	–	–	(80,512)
Upsample2			
conv5	5 × 1, 512	1 × 1	(80,512)
shuffler2	5 × 1, 512	2 × 1	(160,256)
gate5	–	–	(160,256)
conv6	15 × 1, 64	1 × 1	(160,256)

image data. As shown in Fig. 3(a), the regular residual block (ResBlock) adds the input to the output, which has passed through several convolutional, batch normalization and ReLU layers. Each block contains the convolutional layers with 3 × 3 filters and 1 × 1 stride. Utilizing the shortcut connection with no extra parameters, the ResBlock can be defined as

$$y = ReLU[\mathcal{F}(x, W_i) + x], \tag{12}$$



(a) The regular ResBlock. (b) A variant of the ResBlock

Fig. 3. The architecture of ResBlocks. The input x has the shape of (bs, h, w, 2c), where bs, h, w and 2c denote batch size, height, width and channel. (a): the regular block adopted in most ResNets. (b): the "bottleneck" block to reduce computing cost.

where x and y are the input and output, $\mathcal{F}$ and W_i represent the mapping to be learned.

To reduce parameters and improve calculation efficiency, the variant ResBlock is adopted, which comprises a stack of three convolutional layers. As shown in Fig. 3(b), the first 1 × 1 layer reduces the feature dimension to half of the input, giving a smaller map into the 3 × 3 layer. The last 1 × 1 layer restores the feature dimension to that of the original input. The regular ResBlock in Fig. 3(a) uses the full size of the input for convolution, and the weight parameters of training are five times more than those in the variant block.

Table 2 shows the details of the discriminator architecture. Four ResBlocks are stacked and a softmax classifier is used to predict the speaker identity in the final layer.

5. Experiments

5.1. Dataset and basic setup

To evaluate the performance of the proposed approach, we conduct experiments on the Librispeech [46] dataset. Librispeech corpus is an open English speech dataset which is derived from audio books and contains 1000 h speech. We use the train-clean-100 subset in it that contains 251 speakers of 125 females and 126 males. Each speaker is provided with an average of 113 utterances lasting 1–15 s.

All speech data is sampled at 16 kHz and 64-dimensional mel-filter bank coefficients (FBank) are then computed using a sliding hamming window of width 25 ms with 10 ms overlap. To explore the model performance when feeding speech signals of different representations, we also extract mel-frequency cepstral coefficients (MFCC) of dimension 64 with the same window length and overlap. For both FBank and MFCC, each utterance is extracted with 160 frames which form a spectrogram of 160 × 64 size. Mean and variance normalization is used to improve the performance [15]. An energy-based voice activity detection (VAD) [47] is performed at the frame level to remove silence frames. The blank portion is padded with zeros if the utterance has shorter effective region. The same acoustic features are used for the baselines of CNN, LSTM and GAN classifier.

The proposed models are implemented using the machine learning system of tensorflow [48], and trained on the GeForce

Table 2
Discriminator architecture. "Average" denotes the temporal pooling layer. Three ResBlocks are stacked to form a group each time.

Layer name	Structure	Stride	Output dim
conv1	5 × 5, 64	2 × 2	(80,32,64)
ResBlock1	$\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,64 \end{bmatrix} \times 3$	1 × 1	(80,32,64)
conv2	5 × 5, 128	2 × 2	(40,16,128)
ResBlock2	$\begin{bmatrix} 1\times1,64 \\ 3\times3,64 \\ 1\times1,128 \end{bmatrix} \times 3$	1 × 1	(40,16,128)
conv3	5 × 5, 256	2 × 2	(20,8,256)
ResBlock3	$\begin{bmatrix} 1\times1,128 \\ 3\times3,128 \\ 1\times1,256 \end{bmatrix} \times 3$	1 × 1	(20,8,256)
conv4	5 × 5, 512	2 × 2	(10,4,512)
ResBlock4	$\begin{bmatrix} 1\times1,256 \\ 3\times3,256 \\ 1\times1,512 \end{bmatrix} \times 3$	1 × 1	(10,4,512)
Reshape	–	–	(10,2048)
Average	–	–	(2048)
Fully connected	–	–	(512)
softmax	–	–	(251 + 1)

GTX 1080 Ti GPU. Speech samples are shuffled before training. Sixty percentage of all utterances are randomly selected as training data and the rest are used as test data. With a batch size of 64, the network is trained with the Adam optimizer. The training of GANs usually takes a long time to converge. If the test accuracy variation is less than 0.05 in the first 25 epochs, the training lasts for 25 epochs or the training will not stop until 50 epochs. Allowing classification to benefit from the adversarial process, we set $\lambda_1 = 1$ and $\lambda_2 = 1$.

Since GANs are difficult to be trained, some tricks are applied to encourage convergence. To stabilize training, the negative log likelihood objective in L_{adv} is replaced with a least square loss, which is described in Ref. [40]. To get a reliable discriminator in the iteration, the discriminator is trained four times for each generator update. To avoid sparse gradients, ReLUs in the discriminator is replaced with Leaky ReLUs. Label smoothing [29] also contributes to stability improvement. For a real sample, the label 1 is replaced with a random number between 0.7 and 1.0, while the label 0 is replaced with a random number between 0 and 0.3 for a generated sample. In the first 10 epochs,the learning rates for the generator and discriminator are both set as 0.0005 to accelerate the convergence. For the rest of the training, the learning rates are both set as 0.0002.

5.2. Baselines

To evaluate the performance of the proposed approach, we compare it with these four baselines.

I-vector. The i-vector baseline is built on [8] and trained on the Librispeech-100. 60-dimensional feature vectors consisting of 20-dimensional MFCC along with the appended delta and acceleration are extracted, which have been used by many SI researches [49]. They are then reduced to the i-vectors of dimension 200 using probabilistic linear discriminant analysis (PLDA). The DNN is trained for classifying speakers.

X-vector. As reported in [14], with the input of 24-dimensional FBank, the x-vectors are extracted and reduced to dimension 150 using PLDA. The DNN is also used for classification similar to the i-vector system.

CNN. CNNs have been successfully applied in many SI tasks [15,16] and provide superior performance to the traditional i-vector methods. To not only verify the effectiveness of CNN, but also explore the classification performance of the proposed framework eliminating the generation, the baseline of CNN follows the architecture of the discriminator. With the modification to the output layer, the CNN baseline only predicts the speaker identities of the input utterance.

LSTM. The sequential baseline model is built using a 3-layer LSTM, where each has dimension 1024, with a softmax output layer. In order to speed up the convergence of the network, the convolutional layers are introduced before LSTM layers, which reduce the input dimension. A bidirectional LSTM (BiLSTM) based approach is also adopted to make full use of the context information in both forward and backward directions.

6. Experimental results

6.1. Comparison with baselines

For identification under the same condition of input features, evaluations are carried out on the baselines and SpeakerGAN. As described in Table 3, when speech signal is represented as the spectrogram of FBank, the DL based approaches and spakerGAN all achieve lower CER than the i-vector and x-vector systems. The proposed SpeakerGAN achieves a CER of 1.63%, which is 87.60%, 72.70%, 55.83% and 16.84% relatively lower than the baselines of i-vector, x-vector, LSTM+DP and CNN+DP. This table shows that the dropout setting improves the identification accuracy of the standard CNN and LSTM, however contributes little to higher classification accuracy of the proposed method. Meanwhile, the SpeakerGAN obtains an obvious advantage over the isolated standard CNN. This reveals that the generation from the generator and dropout may have the similar effect in preventing neural networks from overfitting and enhancing the network generalization ability.

To verify the necessity of some modules in the SpeakerGAN architecture, we remove them separately and carry out experiments under the same parameter setting. The preprocessing of mean and variance normalization proves to be crucial, which leads to a performance improvement of 87%. There is a considerable drop in performance when VAD is removed from the model. The batch normalization also results in decreasing the CER by 72%, because it reduces the internal covariate shift by controlling the change of the layers' input distributions [50]. Without the Huber loss introduced, the adversarial process becomes unstable and the CER is increased due to the generation of poorer quality. We also test the effect of replacing the condition x with the random noise as the input of the generator, like the basic GAN. In this case, the adversarial network fails to penalize the mismatch between the input and output, and it appears to bring about worse results.

The cross-entropy losses for classification of different systems are given in Table 4. The proposed approach achieves the lowest CER in Table 3, but the loss is higher than CNN and even has small gap with LSTM. This is because the discrimination of real/fake disrupts the classification in training. The discrimination and classification are jointly optimized and they share the same network. Decreasing λ_1 in Eq. (10) can help reduce the loss but may increase the CER. As is known, the smaller cross-entropy loss does not necessarily result in higher identification accuracy, because the network decision depends on the class of the highest probability. Thus, it will not become an issue for the identification task.

6.2. Impact of front-end features

MFCC is the traditional handcrafted feature of the speech signal based on the nonlinear mel scale of frequency. Tirumala et al. [49] have reported that the MFCC-based feature extraction approaches are identified as the most successful and widely used approach for SI feature extraction. MFCC is much easier to model due to the independent components [51], while FBank, instead, is strongly

Table 3
Classification error rate (CER %) of different systems trained on the Librispeech-100. FBank and MFCC are different front-end feature extractions. DP: drop out, BN: batch normalization, norm: mean and variance normalization. The baselines of i-vector and x-vector follow the standard strategy in the original papers and use the same dimension when feeding another input representation.

Methods		CER %	
		FBank	MFCC
Baseline	i-vector	13.15	12.73
	x-vector	5.97	6.14
	CNN	2.73	2.52
	CNN + DP	1.96	2.14
	LSTM	4.25	4.11
	LSTM + DP	3.69	3.38
	BiLSTM	4.12	3.42
SpeakerGAN	**original**	**1.63**	**1.90**
	DP	1.87	1.94
	no BN	4.85	5.22
	no VAD	5.79	5.50
	no norm	9.55	9.85
	no Huber loss	2.00	1.94
	noise input	1.81	1.92

Table 4
Computational losses of different systems. For SpeakerGAN, the classification loss component is given here. i-vec: i-vector. x-vec: x-vector.

Methods	i-vec	x-vec	CNN	LSTM	SpeakerGAN
Loss	0.55	0.24	0.10	0.21	0.18

Table 5
Classification accuracy (ACC %) of SpeakerGAN with different dimensions of FBank and MFCC. The training and test ACC are shown in the table separately. In training, 96 frames are enough for models to give correct prediction, so only 96 frames for each utterance are used here.

Features		Dimension of features				
		8	16	32	48	64
FBank	train	92.19	94.62	97.71	98.75	98.75
	test	88.03	93.01	95.55	97.55	98.14
MFCC	train	89.06	96.42	98.03	97.95	98.54
	test	88.50	95.90	97.55	97.15	97.10

correlated. FBank and MFCC both have the ability to measure the difference of enough resolution among different speakers.

Limited by the model capacity, MFCC and its variations, such as MFCC fusion, are more adopted in the earlier SI systems [8,49]. With the rapid development of neural networks, more and more researches [16,17,20] choose to use FBank as the input representation. The neural networks are capable of modeling the spectral correlations in FBank, thereby can accommodate more useful information and achieve better performance.

To evaluate the performance using different dimensions of different front-end features, FBank and MFCC are used for identification separately. As shown in Table 3, for the networks constructed with convolutional layers, including the CNN and SpeakerGAN, the performance with FBank is better than MFCC on the whole. Conversely, the LSTM based models achieve about 17% lower CER using MFCC. These results suggest that FBank features which contain more complicated hierarchical information suits CNNs better, while LSTM realizes better sequential modeling with MFCC which has been modulated after the discrete cosine transform.

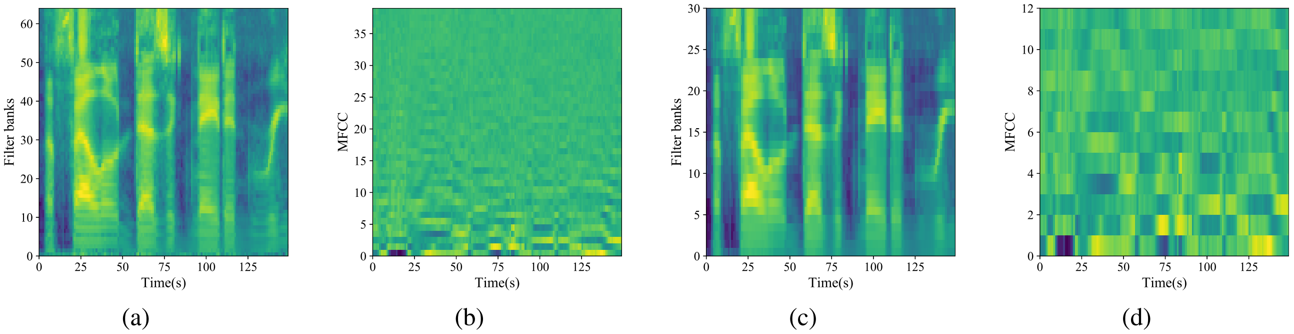
Actually, as shown in Table 5, in the experiments, the increasing dimensionality of FBank usually leads to a higher identification accuracy, while ACC stops increasing even the higher-order MFCC components are extracted. It can be observed from Fig. 4(b) and (d) that the high-order MFCC features present to be of little account because the amplitudes are too low while the first twelve dimensions of MFCC are of high strength. However, the FBanks shown in Fig. 4(a) and (c) carry valid and non-negligible information ranging over the entire banks. It is consistent with the explanation in the previous study [51] that the important information of MFCC is concentrated in the first few features and the discriminant information of FBank is distributed across all coefficients.

6.3. Performance against the amount of training data

Since the SI task suffers from the lack of sufficient speech data for training, we conduct experiments on different scales of the training dataset to verify the effectiveness of the proposed scheme. 10–30 utterances for each speaker are chosen for training and all the remaining utterances are used for test. We select the baselines of the configurations which achieve the best performance in Table 3 and the input feature is FBank. The i-vector and x-vector baselines are not tested here for their high computational cost.

Table 6 reports the ACC achieved with Librispeech-100. The fluctuation range of ACC is recorded to detect the stability trends of SpeakerGAN outputs. It can be observed that with few training data, the SpeakerGAN achieves significant enhancement. With corpus of larger scale, the gap between the baselines and the proposed method is less pronounced. Given only ten utterances each speaker for training, which equate to a non-silence speech clip of 16 s, the SpeakerGAN achieves the ACC of 93.04%, which reduces the CER by 22.44% for the CNN and 28.33% for LSTM. With 20 utterances for training, the SpeakerGAN obtains the ACC of 96.87%, which still has 9.43% and 28.86% advantage over the baselines.

From these results, we can see that the proposed method improves the generalization of the model. Compared with the baselines, with the training data of small scale, our method has alleviated performance degradation resulting from the overfitting problem. The improvement of the proposed method is due to the generated samples produced by the generator network which has learned the distribution of real speech samples. Additionally, the discriminator network simultaneously gains the abilities to classify the input samples and distinguish between the real and fake samples. The distinguishment task that goes on under the adversarial



(a) (b) (c) (d)

Fig. 4. Spectrograms of FBank and MFCC. (a): 64-dimensional FBank. (b): 39-dimensional MFCC. (c): 30-dimensional FBank. (d): 12-dimensional MFCC.

Table 6
ACC (%) of different systems using different scales of datasets. ± indicates the volatility of ACC on the evaluation dataset. Each utterance for training is cropped off silence region, and lasts for about 1.6 s.

Methods	Number of utterances					
	5	10	15	20	25	30
LSTM(DP)	78.68	92.01	92.95	95.60	93.21	95.89
CNN(DP)	80.30	91.46	95.12	96.50	96.87	97.31
SpeakerGAN(original)	85.72 ± 5.90	93.04 ± 4.67	94.42 ± 2.18	96.87 ± 1.18	97.12 ± 0.51	97.54 ± 0.50

Table 7
ACC (%) of DL based methods using different frames for identification. The training and test ACC are shown in the table separately.

Methods		Number of frames						
		8	16	32	64	96	120	152
LSTM(DP)	train	81.90	91.04	94.95	95.70	96.84	97.71	97.71
	test	81.93	90.15	94.32	95.46	94.20	95.35	96.31
CNN(DP)	train	87.59	93.15	95.66	97.71	98.53	98.75	98.75
	test	83.67	92.48	94.71	96.32	96.22	97.55	97.00
SpeakerGAN(original)	train	85.42	92.18	96.44	96.96	98.75	97.71	98.75
	test	82.07	93.33	94.05	96.75	96.55	96.95	98.20

mechanism has introduced new information to amend the original optimization process.

In the training, we found that the classification accuracy of SpeakerGAN fluctuates more than 5% when the corpus is too small. In some cases, the performance improvement obtained by the proposed method is sometimes less obvious even under the scenario of limited training data. We believe this is caused by the instability of the discriminator, which is disturbed by the generator during alternating. When the generator is fixed for more steps in one iteration, the discriminator tends to output consistent results. As the dataset amount increasing, the volatility of ACC is reduced to the normal level. Dataset of larger scale may lead to the generation distribution closer to the target data distribution, resulting in reliable convergence of the discriminator.

6.4. Performance against utterance durations

The duration of utterance is a vital element that affects the SI performance. It can be observed from Table 3 that SpeakerGAN achieves the accuracy of 98.37% using the utterance of fewer than 2 s. This has already gained the superior performance over the state-of-the-art technique [52]. Table 5 suggests that the input size of the spectrogram is more than enough for identifying speakers. Thus, SpeakerGAN appears to be promising for SI under more strict conditions. To evaluate the proposed method with short utterances, different frames of the acoustic features are extracted. The 64-dimensional FBank is extracted to avoid the influence of feature dimension. The same normalization and VAD preprocessing are conducted before extracting.

As shown in Table 7, when eight frames are extracted which correspond to a non-silence speech clip of almost 0.1 s, SpeakerGAN only achieves a training accuracy of 85.42%. Until extracting 96 frames (almost 1 s), the DL models cannot give the correct prediction based on the incomplete information. They all have difficulty distinguishing between different speakers. With the speech segments of fewer than 32 frames, SpeakerGAN obtains superior performance over the LSTM baseline and has a similar performance with the isolated CNN. Given the utterances of longer duration, the proposed method achieves close identification accuracy with the CNN on the test dataset.

7. Conclusions and future work

In this paper, we propose a novel text-independent speaker identification approach based on the CGAN, called SpeakerGAN. It directly utilizes the discriminator as a classifier using the fake samples produced by the generator as the additional class. The Hybrid loss function includes the adversarial loss of the regular GAN, the cross-entropy loss of the labeled data and the Huber loss between the real samples and generation. They are introduced to encourage the diversity of the generated samples and more robust identification. Experimental results demonstrate that SpeakerGAN can achieve higher identification accuracy than other state-of-the-art DL based methods and the traditional i-vector and x-vector systems. Under the constrained condition of limited training data, SpeakerGAN shows obvious superiority over the baselines. Moreover, the proposed method is promising in development of speaker identification with short utterances.

As the future work, SpeakerGAN will be improved for higher stability and lower complexity to be applied in the end-to-end speaker verification task. Moreover, the generated samples from the generator are supposed to be visually understood by adopting other variants of GAN, such as CycleGAN.

CRediT authorship contribution statement

Liyang Chen: Methodology, Writing - original draft. **Yifeng Liu:** Software, Data curation. **Wendong Xiao:** Formal analysis, Writing - review & editing. **Yingxue Wang:** Conceptualization, Investigation. **Haiyong Xie:** Resources, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China under Grants 61673055 and 6167305 and the National Key Research and Development Project (Grant No. 2017YFC0820504).

References

[1] F.K. Soong, A.E. Rosenberg, B. Juang, L.R. Rabiner, Report: a vector quantization approach to speaker recognition, AT&T Technical Journal 66 (2) (1987) 14–26.
[2] T. Matsui, T. Kanno, S. Furui, Speaker recognition using hmm composition in noisy environments, Computer Speech & Language 10 (2) (1996) 107–116.

[3] D.A. Reynolds, R.C. Rose, Robust text-independent speaker identification using gaussian mixture speaker models, IEEE Transactions on Speech and Audio Processing 3 (1) (1995) 72–83.

[4] D.A. Reynolds, T.F. Quatieri, R.B. Dunn, Speaker verification using adapted gaussian mixture models, Digital Signal Processing 10 (1) (2000) 19–41.

[5] W.M. Campbell, D.E. Sturim, D.A. Reynolds, Support vector machines using gmm supervectors for speaker verification, IEEE Signal Processing Letters 13 (5) (2006) 308–311.

[6] A. Solomonoff, W.M. Campbell, I. Boardman, Advances in channel compensation for svm speaker recognition, in: Proceedings. IEEE International Conference on Acoustics, Speech, and Signal Processing., vol. 1, 2005, pp. I/629–I/632..

[7] P. Kenny, Joint factor analysis of speaker and session variability: theory and algorithms, CRIM, Montreal, (Report) CRIM-06/08-13, vol. 14, pp. 28–29, 2005..

[8] N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel, P. Ouellet, Front-end factor analysis for speaker verification, IEEE Transactions on Audio, Speech, and Language Processing 19 (4) (2011) 788–798.

[9] Y. Lei, N. Scheffer, L. Ferrer, M. McLaren, A novel scheme for speaker recognition using a phonetically-aware deep neural network, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) , 2014, pp. 1695–1699.

[10] E. Variani, X. Lei, E. McDermott, I.L. Moreno, J. Gonzalez-Dominguez, Deep neural networks for small footprint text-dependent speaker verification, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014, pp. 4052–4056.

[11] G. Heigold, I. Moreno, S. Bengio, N. Shazeer, End-to-end text-dependent speaker verification, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016, pp. 5115–5119.

[12] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, S. Khudanpur, Deep neural network-based speaker embeddings for end-to-end speaker verification, in: IEEE Spoken Language Technology Workshop (SLT), 2016, pp. 165–170.

[13] D. Snyder, D. Garcia-Romero, D. Povey, S. Khudanpur, Deep neural network embeddings for text-independent speaker verification, in: Proc. Interspeech, 2017, pp. 999–1003..

[14] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, S. Khudanpur, X-vectors: Robust dnn embeddings for speaker recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 5329–5333.

[15] A. Nagrani, J.S. Chung, A. Zisserman, Voxceleb: a large-scale speaker identification dataset, arXiv preprint arXiv:1706.08612, 2017..

[16] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, Z. Zhu, Deep speaker: an end-to-end neural speaker embedding system, arXiv preprint arXiv:1705.02304, 2017..

[17] J.S. Chung, A. Nagrani, A. Zisserman, Voxceleb2: Deep speaker recognition, in: Proc. Interspeech, 2018, pp. 1086–1090..

[18] W. Xie, A. Nagrani, J.S. Chung, A. Zisserman, Utterance-level aggregation for speaker recognition in the wild, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 5791–5795.

[19] P.S. Nidadavolu, V. Iglesias, J. Villalba, N. Dehak, Investigation on neural bandwidth extension of telephone speech for improved speaker recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 6111–6115.

[20] T. Bian, F. Chen, L. Xu, Self-attention based speaker recognition using cluster-range loss, Neurocomputing 368 (2019) 59–68.

[21] C. Chen, S. Zhang, C. Yeh, J. Wang, T. Wang, C. Huang, Speaker characterization using tdnn-lstm based speaker embedding, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 6211–6215.

[22] F.A. Rezaur rahman Chowdhury, Q. Wang, I.L. Moreno, L. Wan, Attention-based models for text-dependent speaker verification, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 5359–5363..

[23] M. Ravanelli, Y. Bengio, Speaker recognition from raw waveform with sincnet, in: IEEE Spoken Language Technology Workshop (SLT), 2018, pp. 1021–1028.

[24] N. Adiga, Y. Pantazis, V. Tsiaras, Y. Stylianou, Speech enhancement for noise-robust speech synthesis using wasserstein gan, in: Proc. Interspeech, 2019, pp. 1821–1825..

[25] T. Kaneko, H. Kameoka, Cyclegan-vc: Non-parallel voice conversion using cycle-consistent adversarial networks, in: 26th European Signal Processing Conference (EUSIPCO), 2018, pp. 2100–2104.

[26] T. Kaneko, H. Kameoka, K. Tanaka, N. Hojo, Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 6820–6824.

[27] D. Paul, Y. Pantazis, Y. Stylianou, Non-parallel voice conversion using weighted generative adversarial networks, in: Proc. Interspeech, 2019, pp. 659–663..

[28] Y. Hono, K. Hashimoto, K. Oura, Y. Nankaku, K. Tokuda, Singing voice synthesis based on generative adversarial networks, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 6955–6959.

[29] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, X. Chen, Improved techniques for training gans, Advances in Neural Information Processing Systems 29 (2016) 2234–2242.

[30] J.T. Springenberg, Unsupervised and semi-supervised learning with categorical generative adversarial networks, arXiv preprint arXiv:1511.06390, 2015..

[31] A. Odena, Semi-supervised learning with generative adversarial networks, arXiv preprint arXiv:1606.01583, 2016..

[32] P. Shen, X. Lu, S. Li, H. Kawai, Conditional generative adversarial nets classifier for spoken language identification, in: Proc. Interspeech, 2017, pp. 2814–2818..

[33] X. Miao, I. McLoughlin, S. Yao, Y. Yan, Improved conditional generative adversarial net classification for spoken language recognition, in: IEEE Spoken Language Technology Workshop (SLT), 2018, pp. 98–104.

[34] G. Bhattacharya, J. Monteiro, J. Alam, P. Kenny, Generative adversarial speaker embedding networks for domain robust end-to-end speaker verification, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 6226–6230.

[35] P.S. Nidadavolu, S. Kataria, J. Villalba, N. Dehak, Low-resource domain adaptation for speaker recognition using cycle-gans, arXiv preprint arXiv:1910.11909, 2019..

[36] J. Zhang, N. Inoue, K. Shinoda, I-vector transformation using conditional generative adversarial networks for short utterance speaker verification, in: Proc. Interspeech, 2018, pp. 3613–3617..

[37] D. Michelsanti, Z.-H. Tan, Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification, in: Proc. Interspeech, 2017, pp. 2008–2012..

[38] W. Ding, L. He, Mtgan: Speaker verification through multitasking triplet generative adversarial networks, in: Proc. Interspeech, 2018, pp. 3633–3637..

[39] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Advances in Neural Information Processing Systems 27 (2014) 2672–2680.

[40] X. Mao, Q. Li, H. Xie, R.Y. Lau, Z. Wang, S. Paul Smolley, Least squares generative adversarial networks, in: IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2794–2802.

[41] M. Kaur, S. Satapathy, R. Soundrapandiyan, J. Singh, Targeted style transfer using cycle consistent generative adversarial networks with quantitative analysis of different loss functions, International Journal of Knowledge-based and Intelligent Engineering Systems 22 (4) (2018) 239–247.

[42] F. Fang, J. Yamagishi, I. Echizen, J. Lorenzo-Trueba, High-quality nonparallel voice conversion based on cycle-consistent adversarial network, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 5279–5283.

[43] Y.N. Dauphin, A. Fan, M. Auli, D. Grangier, Language modeling with gated convolutional networks, in: Proceedings of the 34th International Conference on Machine Learning, ser. ICML'17, vol. 70, 2017, p. 933–941..

[44] W. Shi, J. Caballero, F. Huszár, J. Totz, A.P. Aitken, R. Bishop, D. Rueckert, Z. Wang, Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1874–1883.

[45] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[46] V. Panayotov, G. Chen, D. Povey, S. Khudanpur, Librispeech: An asr corpus based on public domain audio books, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 5206–5210.

[47] Jing Pang, Spectrum energy based voice activity detection, in: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), 2017, pp. 1–5.

[48] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: A system for large-scale machine learning, in: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, 2016, pp. 265–283.

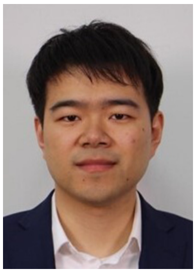
[49] S.S. Tirumala, S.R. Shahamiri, A.S. Garhwal, R. Wang, Speaker identification features extraction methods: a systematic review, Expert Systems with Applications 90 (2017) 250–271.

[50] S. Santurkar, D. Tsipras, A. Ilyas, A. Madry, How does batch normalization help optimization? Advances in Neural Information Processing Systems 31 (2018) 2483–2493..

[51] A. Mohamed, G. Hinton, G. Penn, Understanding how deep belief networks perform acoustic modelling, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012, pp. 4273–4276.

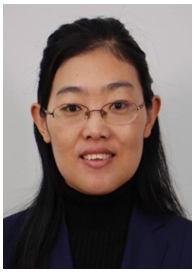
[52] C. Zhang, W. Chen, C. Xu, Depthwise separable convolutions for short utterance speaker identification, in: IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), 2019, pp. 962–966.

Liyang Chen received the B.E. degree from the School of Automation & Electrical Engineering, University of Science and Technology Beijing, PR China, in 2018. He is currently pursuing the master degree in Control Theory and Control Engineering at University of Science and Technology Beijing, PR China. His research interests include machine learning, deep learning, and speech signal processing.



Yifeng Liu received the B.S. degree in Electronic Engineering from Jianghan University, Wuhan, China, in 2011 and the Ph.D. degrees in Electronic Engineering from Wuhan University, Wuhan, China, in 2016. He is currently a senior engineer of China Academy of Electronics and Information Technology, and the research assistant of academician workstation at National Engineering Laboratory for Public Safety Risk Perception and Control by Big Data (PSRPC), Beijing, China. His current research interests include deep learning, machine learning, computer vision, and knowledge engineering.

Wendong Xiao received the B.S. and Ph.D. degrees from Northeastern University, Shenyang, China, in 1990 and 1995, respectively. He held various academic and research positions with Northeastern University, POSCO Technical Research Laboratories, South Korea, Nanyang Technological University, Singapore, and the Institute for Infocomm Research, Agency for Science, Technology and Research (A* STAR), Singapore. He is currently a Professor with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China. His current research interests include information fusion, big data processing, wireless localization and tracking, energy harvesting based resource management, wireless sensor networks, and Internet of Things.



Yingxue Wang received the Ph.D. degrees in Beijing Institute of Technology, Beijing, China, in 2017. She is currently a senior engineer of China Academy of Electronics and Information Technology, and the research assistant of academician workstation at National Engineering Laboratory for Public Safety Risk Perception and Control by Big Data (PSRPC), Beijing, China. Her current research interests include machine learning, speech signal processing and computer vision.

Haiyong Xie received the B.S. degree from University of Science and Technology of China (USTC), Hefei, China, in 1997, and the M.S. and Ph.D. degrees in computer science from Yale University, in 2005 and 2008, respectively. He is the Director for the Innovation Center, China Academy of Electronics and Information Technology, the Vice-Director for National Engineering Laboratory for Public Safety Risk Perception and Control by Big Data (PSRPC), Beijing, China, and a Professor with the School of Computer Science and Engineering, USTC. His research interest includes artificial intelligence, network traffic engineering, enterprise network traffic optimization, software-defined networking, and future Internet architectures.