Text-Independent Speaker Verification Based on Triplet Convolutional Neural Network Embeddings

Article *in* IEEE/ACM Transactions on Audio Speech and Language Processing · April 2018

DOI: 10.1109/TASLP.2018.2831456

3 authors:

Chunlei Zhang
Tencent
67 PUBLICATIONS **1,554** CITATIONS
SEE PROFILE

Kazuhito Koishida
Microsoft
70 PUBLICATIONS **1,593** CITATIONS
SEE PROFILE

John H. L. Hansen
The University of Texas at Dallas
715 PUBLICATIONS **15,912** CITATIONS
SEE PROFILE

Text-Independent Speaker Verification Based on Triplet Convolutional Neural Network Embeddings

Chunlei Zhang, *Student Member, IEEE*, Kazuhito Koishida, *Member, IEEE*, and John H. L. Hansen, *Fellow, IEEE*

***Abstract*—The effectiveness of introducing deep neural networks into conventional speaker recognition pipelines has been broadly shown to benefit system performance. A noveltext-independent speaker verification (SV) framework based on the triplet loss and a very deep convolutional neuralnetwork architecture (i.e., Inception-Resnet-v1) are investigated in this study, where a fixed-length speaker discriminative embedding is learned from sparse speech features and utilized as a feature representation for the SV tasks. A concise description of the neural network based speaker discriminative training with triplet loss is presented. An Euclidean distance similarity metric is applied in both network training and SV testing, which ensures the SV system to follow an end-to-end fashion. By replacing the final max/average pooling layer with a spatial pyramid pooling layer in the Inception-Resnet-v1 architecture, the fixed-length input constraint is relaxed and an obvious performance gain is achieved compared with the fixed-length input speaker embedding system. For datasets with more severe training/test condition mismatches, the probabilistic linear discriminant analysis (PLDA) back end is further introduced to replace the distance based scoring for the proposed speaker embedding system. Thus, we reconstruct the SV task with a neural network based front-end speaker embedding system and a PLDA that provides channel and noise variabilities compensation in the back end. Extensive experiments are conducted to provide useful hints that lead to a better testing performance. Comparison with the state-of-the-art SV frameworks on three public datasets (i.e., a prompt speech corpus, a conversational speech Switchboard corpus, and NIST SRE10 10 s–10 s condition) justifies the effectiveness of our proposed speaker embedding system.**

***Index Terms*—Speaker recognition, very deep convolutional neutral networks, i-vector, PLDA, triplet loss, spatial pyramid pooling.**

I. Introduction

SPEAKER verification (SV) is a binary classification problem which aims to verify a claimed identify based on the claimed/enrolled speaker model. According to different application scenarios, speaker verification systems fall into two categories: *text-dependent* and *text-independent* [1], [2].

The text-dependent SV scenario requires the same set of text phrases for enrollment and test. Combined with a keyword spotting system (KWS), text-dependent SV can be integrated within an intelligent personal assistants such as Microsoft Cortana, Apple Siri, Google Home etc., where KWS and text-dependent SV serves as a keyword voice-authenticated wake-up to enable subsequent voice interaction [3]–[5]. Recent advancements in text-dependent SV have been reported using deep neural networks (DNNs) and recurrent neural networks (RNNs) for speaker discriminative or phonetic discriminative network training, where intermediate frame-level features such as d-vectors [3], [5], bottleneck activations or phonetic alignments are extracted to formulate utterance-level speaker representations [6], [7]. More recently, DNNs, RNNs and convolution neural networks (CNNs) with an end-to-end loss $\log p(accept/reject)$ have been investigated to discriminate between the same-speaker and different-speaker pairs for global keyword (e.g., "OK Google" and "Hey Cortana") speaker verification tasks, and shown to achieve better performance compared with conventional techniques such as GMM-UBM or i-Vector/PLDA [3], [4]. For these end-to-end systems, the impressive performances can be attributed to: a) a large dataset with more than 10k+ speakers, which means sufficient variabilities have been introduced in the speaker discriminative network training; b) text-dependent speaker verification with a fixed lexicon, where phonetic variability is largely constrained. However, it should be noted that these systems have not successfully been applied to text-independent speaker verification.

In the context of text-independent speaker verification, the i-Vector/probabilistic linear discriminant analysis (PLDA) framework and its variants are the state-of-the-art across many tasks [8], [9]. The i-Vector framework learns a single low-dimensional subspace called the total variability subspace, through which utterances of variable-length can be represented as fixed-length feature vectors [10]. Despite great successes achieved in those evaluations, i-Vector systems are prone to have performance degradation when enrollment/test utterance durations are short. Table I shows EERs with respect to different test utterance du-

TABLE I
I-Vector Based System Performance Versus Different Durations in SRE16 (EER %)

	10s	20s	30s	40s	50s	60s&up
EER	12.11	11.51	9.23	8.13	8.11	7.42

Manuscript received September 28, 2017; revised February 14, 2018 and April 10, 2018; accepted April 17, 2018. Date of publication April 30, 2018; date of current version June 1, 2018. This work was supported in part by the AFRL under contract FA8750-15-1-0205 and in part by the University of Texas at Dallas from the Distinguished University Chair in Telecommunications Engineering. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Tomi Kinnunen. *(Corresponding author: John H. L. Hansen.)*

C. Zhang and J. H. L. Hansen are with the Center for Robust Speech Systems, Erik Jonsson School of Engineering and Computer Science, The University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: chunlei.zhang@utdallas.edu; john.hansen@utdallas.edu).

K. Koishida is with the Microsoft Corporation, Redmond, WA 98052 USA (e-mail: kazukoi@microsoft.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2018.2831456

rations in CRSS submissions for NIST SRE16 [11]. A clear speaker verification performance degradation can be observed in this analysis. Also, early NIST SREs simplified the speaker recognition problem being ensure that the duration of enrollment and test utterances are constrained in many evaluation conditions. As a result, relatively less attention was paid to address the variability in utterance durations, especially when short duration utterances are experienced in practical scenarios.

To compensate for insufficient information or context mismatch due to short duration in i-Vector based text-independent SV, several techniques have been proposed in recent studies. In [12], the authors propagate the uncertainty from i-Vector estimation into PLDA modeling for speaker verification, and show substantial performance improvement on the NIST SRE10 core and extended core conditions where duration variability is introduced by randomly truncating the enrollment/test utterances in the evaluation trials. Hasan *et al.* [13] proposed to employ log-scale duration information in the score calibration for duration mismatch compensation, which also can be viewed as providing uncertainty for short utterances in the score level. In addition to these methods, replacing the UBM posteriors with more supervised phonetic DNN posteriors at the acoustic model level can also be beneficial for short utterance in general [7], [14]. However, the advancement comes at a cost of greatly increased computational complexity and demanding well-annotated data, and the performance gain is mostly limited to English data [8], [11].

When we recall the traditional GMM-UBM based methods (including supervector, Joint Factor Analysis and i-Vector), maximum a posteriori (MAP) estimation has been the key step that adapts the universal background means to speaker dependent feature vectors [10], [15]–[17]. For short duration conditions, since MAP adaptation is performed based on limited data, the adapted Gaussian means are very sparse and lead to relatively poor speaker recognition performance.

Using different deep learning frameworks with end-to-end loss functions to train speaker discriminative embeddings has drawn more attention recently. Snyder *et al.* and Garcia *et al.* [18], [19] have shown that deep neural networks with an end-to-end similarity metric or DNN based speaker embedding could outperform the i-Vector baselines. Competitive performances have been reported with speaker embedding systems based on triplet loss function in either speaker diarization or speaker verification tasks [20]. We have also proposed to apply fixed-dimensional spectrogram as the input to Inception-resnet-v1 for speaker embedding extraction [21], [22]. In that study, the triplet loss is employed to optimize the network training [23], where the $Euclidean\ distance$ is used in both training and test phase, thus the entire SV system is developed in an end-to-end fashion. From the results reported in [18], [22], end-to-end systems achieved better performance compared with the i-Vector/PLDA frameworks, especially when utterances are short.

Addressing variable-length input remains an interesting topic for deep neural network based speaker embedding system [18], [19], [20], [24], [25]. Our most recent work investigates the construction of an end-to-end system which has flexibility in utterance duration [26]. Previously in [22], to make a fixed length input to the network (e.g., 4 s), we performed cropping if the utterance is longer than 4 s. To retrieve discarded information for long utterances, we upgrade the network architecture so that arbitrary length utterances can be directly mapped into fixed-length speaker embeddings. More specifically, a Spatial Pyramid Pooling (SPP) layer is proposed to replace the final average pooling layer within the Inception-resnet-v1 architecture [27]. The modified Inception-resnet-v1 network would process variable-length speech segments while producing fixed-length speaker embeddings as the output.

In this current study, a systematic investigation is conducted based on triplet convolutional neural network speaker embedding system [22], [26], with more insights for more challenging datasets. First, we present an overview of the components which are essential for the proposed speaker embedding system, including the concept of triplet, triplet loss, triplet sampling/selection, Inception-resnet-v1 architecture and the score metric. Next, two approaches which address variable-length inputs are described as an extension of our fixed-length end-to-end system. To utilize the advancement of SV back-end classifiers for modeling channel and noise variability [28], [29], the end-to-end system is further separated into two parts: a very deep CNN to produce the speaker embeddings and an independent classifier to distinguish between same-speaker target trials and different-speaker nontarget trials. We perform speaker verification experiments on three corpora: a prompt speech corpus, a more challenging conversational speech Switchboard corpus [30] and NIST SRE10 10 s–10 s condition. The former two datasets are originally collected for speech recognition and therefore segmented into short duration utterances, while the last NIST SRE10 10 s–10 s is a standard evaluation protocol which was design to evaluate the SV system performance on short duration utterances. Since the main focus of this study is for short utterance text-independent speaker verification, we believe that the experiments on these three corpora should justify the technical and statistical soundness of the proposed SV framework.

Although more detailed explanations and analysis can be found throughout this paper, let us first summarize the core contributions here:

1) We provide a novel speaker embedding framework for text-independent speaker verification based on deep networks and triplet loss. The proposed method outperforms state-of-the-art i-Vector/PLDA solutions in various evaluations, especially for short duration utterances;

2) The speaker embedding approach results in considerably simplified SV systems, compared with traditional i-Vector/PLDA methods;

3) The very deep CNN architecture is modified with a SPP layer for variable-length input, which could potentially be applied to general sequential data in additional to speech.

4) We provide an experimental evaluation that conventional back-end classifiers combined with the proposed triplet convolutional neural network speaker embeddings can further improve SV performance with a big margin.

The remainder of this paper is organized as follows. An overview of the end-to-end framework is presented in Section II. The methods which handle variable-length utterances are described in Section III. The corpora together with corresponding baselines used for system development are introduced in

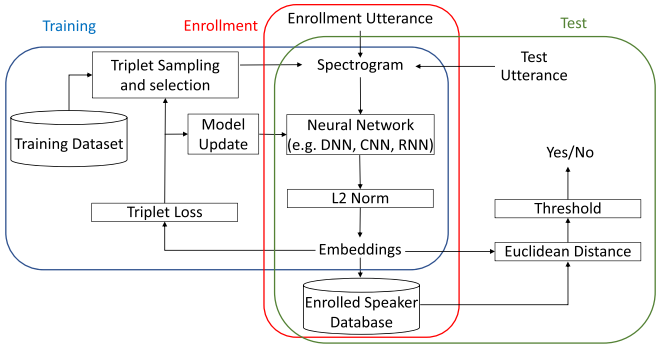

Fig. 1. The architecture of our end-to-end triplet loss based system for text-independent speaker verification.

Section IV. Section V details the experimental results with the proposed systems for different datasets, as well as how these systems behave in contrast with the i-Vector/PLDA baseline. Finally, we conclude our work in Section VI.

II. End-to-End Speaker Verification System

This section describes an overview architecture of our proposed end-to-end speaker verification. The details of its essential components and modifications for speaker embedding network training are presented in the following sections.

A. Overview of System Structure

The main idea behind the end-to-end system is depicted in Fig. 1. For the speaker discriminative embedding training, a *triplet sampling* module samples a batch of triplets so that each triplet consists of an *anchor* x^a as a reference utterance, a *positive* x^p which is an utterance from the same speaker with *anchor*, and a *negative* x^n which is from a different speaker. We propose a deep architecture f_θ (can be flexible to apply many different deep nets, Inception-resnet-v1 in our study) which maps the acoustic features x into the fixed length embeddings $f_\theta(x) \in \mathbb{R}^d$. The objective of the network training is to minimize the *distance* between the embeddings of the *anchor* and *positive* samples, while maximizing the *distance* between the embeddings of the *anchor* and *negative* samples. The L_2 normalization constrains the speaker embedding into a unit hypersphere such that the d-dimensional feature vector satisfies the constrain to $\|f_\theta(x)\|_2 = 1$. Here, the L_2-Norm can be viewed as replacement of length-normalization for i-Vector based SV systems [31].

A similarity metric called the *triplet loss* is employed to optimize networking training, where the network parameter θ is updated after each batch of triplets.

B. Triplet Loss

Triplet loss was originally proposed in [23] for learning discriminative face embeddings from images. For speaker verification, we also want the anchor embeddings $f(x_i^a)$ to be more similar to the positive embeddings $f(x_i^p)$ than to any negative embeddings $f(x_i^n)$, (i.e., network training wants the embeddings to satisfy the following relation):

$$\|f(x_i^a) - f(x_i^p)\|_2^2 < \|f(x_i^a) - f(x_i^n)\|_2^2, \quad \forall (x_i^a, x_i^p, x_i^n) \in \mathcal{T} \tag{1}$$

where $\mathcal{T}$ is the batch of triplets, with (x_i^a, x_i^p, x_i^n) representing a single triplet. A margin α is empirically defined such that a sufficient distance is enforced between the positive and negative pairs with network mapping. Here, we employ the *Euclidean distance* as the similarity criteria. The triplet loss is formulated as (3) with the objective to minimize this loss over the batch $\mathcal{T}$ of triplets:

$$\Delta_i = \|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha, \tag{2}$$

$$L = \sum_{i=1}^{N} \max(0, \Delta_i), (x_i^a, x_i^p, x_i^n) \in \mathcal{T} \tag{3}$$

where L is the triplet loss over a mini-batch, N is the batch size, the gradient w.r.t the "anchor" input f_θ^a, "positive" input f_θ^p, and "negative input f_θ^n:

$$\frac{\partial L}{\partial f_\theta^a} = \sum_{i=1}^{N} \begin{cases} 2(f(x_i^n) - f(x_i^p)), & \text{if } \Delta_i \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

$$\frac{\partial L}{\partial f_\theta^p} = \sum_{i=1}^{N} \begin{cases} 2(f(x_i^p) - f(x_i^a)), & \text{if } \Delta_i \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

$$\frac{\partial L}{\partial f_\theta^n} = \sum_{i=1}^{N} \begin{cases} 2(f(x_i^a) - f(x_i^n)), & \text{if } \Delta_i \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

With a hinge loss like design in (3), triplet samples which are already well separated (corresponding gradient is 0) will not contribute to the gradient calculation for the batch-wise network update according to (4)–(6), which speeds up the learning process during training.

C. Triplet Sampling and Selection

Similar to the triplet sampling strategy proposed in [20] and [23], we select triplets which violate the constraint $\|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2$, with empirical margin $\alpha = 0.2$.

In this study, one epoch will not see all the training speakers (M speakers) due to triplet sampling. Instead, n segments are randomly sampled from each of the m speakers from the training speaker pool, this leads to a total of $mn(n-1)/2$ anchor-positive pairs. Then, for each of those pairs, we randomly choose one negative sample out from all $(m-1)n$ negative candidates. This operation results in $mn(n-1)/2$ triplets for one epoch. It is noted that one *epoch* only samples a small subset of training speakers (i.e., $m \ll M$). The triplet sampling strategy can be viewed as statistical version of sampling all the training speakers, but in a more efficient way. One can observe performance on the validation set to better monitor the training process between these "shrinked" epochs.

The actual speaker number m and segments number n of each speaker depends on different datasets. All the detailed setup for each dataset can be found in Section V-A.

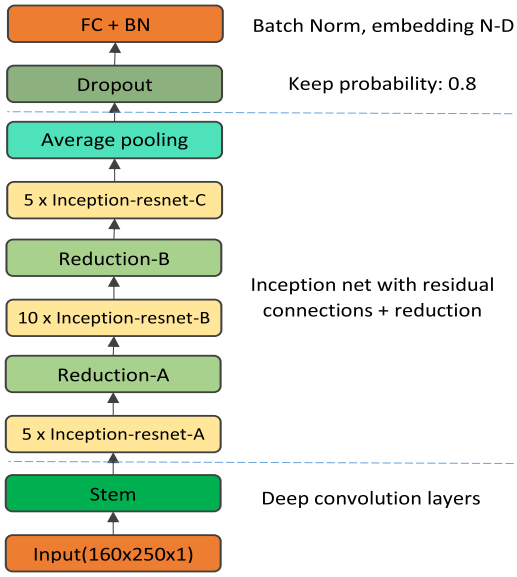

Fig. 2. A simplified architecture of Inception-Resnet-v1 network. The $Stem$ is a particular convolutional network module before the Inception-resnet blocks, detailed implementation can be found in [32].

D. Inception-Resnet-v1 Network

The network architecture proposed for our speaker embedding system training is Inception-resnet-v1, which has shown to be the state-of-the-art framework for image classification tasks in the computer vision community [32]. It is an extension of Inception net with residual connections added to overcome the problem of vanishing/exploding gradients, which is a very common problem in very deep neural network architectures [33], [34]. The network will output 1792 feature maps at the final convolutional layer, which behaves like a UBM model in conventional SV systems (i.e., conventional SV pipelines adopt UBM models to perform acoustic feature alignment, while our proposed architecture uses a fixed number of filters to achieve the same purpose). Fig. 2 is a simplified diagram of the Inception-resnet-v1 network, where more details about this very deep CNN based network architecture can be found in [32]. It should be noted that the Inception-resnet-v1 is a hand-craft design with only one hyperparameter which needs to be tuned: the embedding size controlled by the final fully connected layer. Also, different network architectures can be applied to our end-to-end system, such as Inception network and Bi-LSTM which are already proved to be effective in similar tasks [20], [23].

E. Speaker Verification Evaluation

Through the speaker discriminative network training, the extracted speaker embedding can be used to measure the similarity between speakers. In this phase, the negative Euclidean distance between pairs is employed as the similarity score as (7).

$$S(x_{\text{enroll}}, x_{\text{test}}) = -\|f(x_{\text{enroll}}) - f(x_{\text{test}})\|_2^2, \tag{7}$$

where $\|\cdots\|_2$ is the 2-norm operation of a vector. With this score metric employed as the back-end, one can consider the SV system follows an end-to-end paradigm without additional training of classifiers on top of speaker embeddings.

Here, we established a connection between (7) and commonly used Cosine Distance Scoring (CDS) method in speaker verification. For an L_2-normalized vector $\mathbf{x}$, $\mathbf{y}$ (i.e., $\|\mathbf{x}\|_2 = 1$, $\|\mathbf{y}\|_2 = 1$), it is shown that negative squared Euclidean distance is proportional to the cosine distance:

$$\begin{aligned} -\|\mathbf{x}-\mathbf{y}\|_2^2 &= -(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y}) \\ &= -\mathbf{x}^T\mathbf{x} + 2\mathbf{x}^T\mathbf{y} - \mathbf{y}^T\mathbf{y} \\ &= 2\cos\angle(\mathbf{x},\mathbf{y}) - 2 \end{aligned} \tag{8}$$

Equation (8) indicates that triplet speaker embedding with cosine distance is actually the end-to-end system with negative Euclidean distance. For this consideration, the "triplet_embedding+CDS" system in Fig. 12 for example is the equivalent to the end-to-end system.

To further improve the SV robustness against channel and noise variabilities, it is still possible to utilize previous developed back-end classifiers from speaker verification community, (e.g., replacing i-Vectors with the utterance level speaker embeddings for text-independent speaker verification). We show that conventional back-ends such as PLDA can further improve SV performance in Section V.

III. System Extension for Variable-Length Input

To ensure the speaker embedding system to be flexible against utterance duration, two approaches are investigated: 1) truncate the variable length utterances into multiple fixed size segments, and apply the model developed for fixed-length input; 2) modify the network architecture such that the end-to-end system can process variable length utterances. With additional information added for SV decision making, it is expected that these extensions could lead to improved performance for long utterances with consistent accuracy retained on short utterances.

A. Incremental Speaker Embedding Average

It has already been shown that multiple utterances for speaker enrollment or multi-session SV improves overall speaker verification performance [35]. To make full use of the long utterances instead of discarding them, the long duration utterances are cut into fixed length segments, followed by a feed-forward pass of the fixed-length model to extract speaker embeddings, and then average the speaker embeddings for an utterance-level based verification process. Fig. 3 is a flow diagram for this "incremental" speaker embedding average operation, where a skip rate of 0.5 is used here (e.g. 2 sec for a 4-sec system).

B. Inception-Resnet-v1 With Spatial Pyramid Pooling

An alternative way to handle variable-length duration audio files is to modify the network architecture such that the network can directly produce fixed-length feature vectors before the fully connected layer. In this aspect, an RNN is one suitable architecture for sequential speech utterances [20], [36]. Also, developing convolutional network architectures that can han-

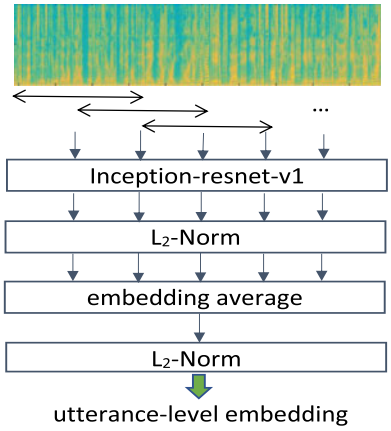

Fig. 3. Flow diagram of incremental speaker embedding average operation, where a mean speaker embedding is produced for the next stage in speaker verification.

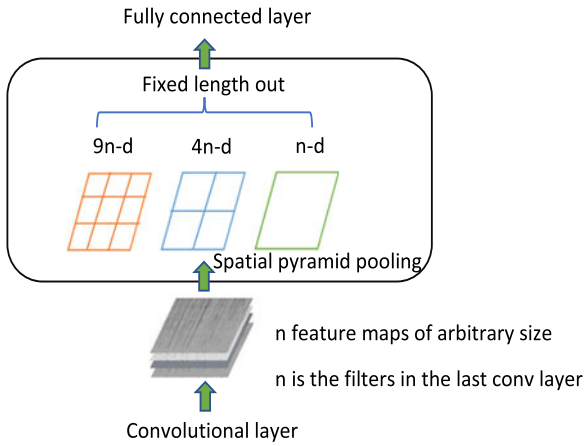

Fig. 4. Spatial pyramid pooling operation as a technique to replace average pooling for extracting speaker embeddings.

dle variable-length features is still an interesting direction in deep learning community [27], [37]. Since the current preliminary system with Inception-resnet-v1 has already been shown to be effective in text-independent speaker verification tasks [22], it is more attractive to incorporate a technique which handles variable-length utterances with the alternative CNN-based network. It is believed that Spatial Pyramid Pooling [27] can be an alternative solution for this purpose. Fig. 4 shows the fundamental structure of the Spatial Pyramid Pooling operation.

Instead of employing a sliding window to max/average pool the feature maps of the traditional convolutional layer output, where the number of the sliding window depends on the input size, spatial pyramid pooling can be used in order to maintain spatial information by pooling in local spatial bins. These spatial bins have sizes proportional to the actual input feature size, so the number of bins is fixed regardless of the image size. As illustrated in Fig. 4, these feature maps are divided into 1×1, 2×2, 3×3 small patches, followed by average pooling performed over these patches, which results in an output fixed-length vector as the input to the following fully connected layer.

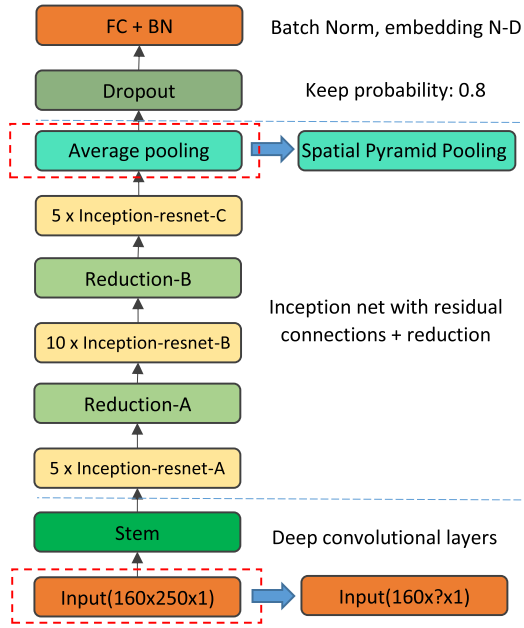

Fig. 5. Inception-Resnet-v1 network architecture with spatial pyramid pooling.

As shown in right half of Fig. 5, the final average pooling layer is replaced with the spatial pyramid pooling layer. Note that with Inception-resnet-v1, the feature maps after the final convolutional layer are usually of small size, where in the current SPP implementation, it is only necessary to apply 1×1 and 2×2 spatial division. Also, for training speed consideration, zero-padding is performed for the samples of the same data batch.

IV. Evaluation Corpora and Baseline Systems for Speaker Verification

Three corpora are used to evaluate SV performance of the proposed methods. *Corpus 1* contains only prompted speech,[1] *Corpus 2* is the Switchboard ASR corpus with conversational speech [30], *Corpus 3* is NIST SRE10 10 s–10 s. All three datasets are publicly available and evaluated in the short duration format. The statistical details of three corpora and corresponding baselines are provided in the following sections.

A. Corpus 1 and the Baseline Systems

Corpus 1 is a large collection of speakers consisting of recordings from three different mainstream platforms, (i.e., Android, iPhone and Windows Phone). The corpus was split for network training, system validation and final evaluation, without speaker overlap among the subsets, see Table II for corpus statistics. There is a total 2790 speakers in the corpus, with approximately 300 short utterances for each speaker. The duration distribution are illustrated in Fig. 6 with a mean duration of 4 s.

To validate and monitor system training, 180 speakers were selected from the validation set. 20 utterance is randomly selected from each speaker, which results in 190 target and 179

[1]http://kingline.speechocean.com/exchange.php?id=1191&act=view

TABLE II
THE NUMBER OF SPEAKERS IN TRAINING, DEVELOPMENT AND TEST SETS, AND CORRESPONDING MEAN UTTERANCE DURATION STATISTICS

	Android	iPhone	WinPhone	total	mean/s
training	954	470	249	1673	4.02
validation	318	156	83	557	3.98
test	319	158	83	560	3.97

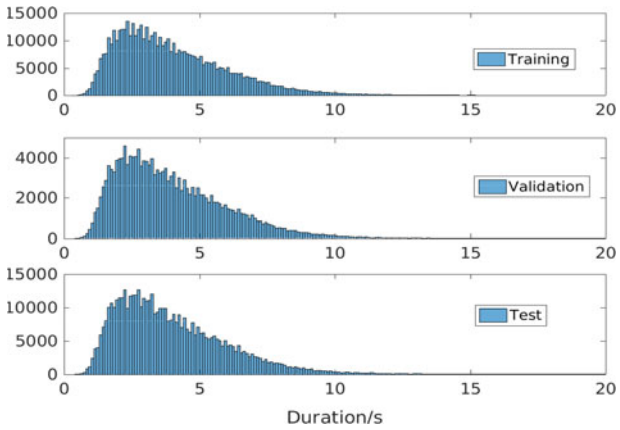

Fig. 6. Duration distributions of training, validation, test set.

nontarget trials per speaker. For system performance evaluation, 450 speakers are picked from the test pool. For each speaker, 10 utterances are sampled as the enrollment data, 80 target trials and 80 nontarget trials are generated to keep the evaluation balanced, where test utterance durations are in the range 0.5–27 s. In total, 720K trials are created for testing.

The i-Vector system developed is based on the Kaldi SRE10/v1 [38]. Front-end features consist of 20 MFCCs with a frame-length of 30ms that are mean-normalized over a sliding window of up to 3 seconds. Delta and acceleration features are appended to create 60 dimensional feature vectors. Non-speech portions of the utterances are removed with energy based voice activity detection. The UBM is a 1024 component full-covariance GMM. The system uses a 400 dimension i-Vector extractor. Prior to PLDA scoring, i-Vectors are both centered and length normalized. The entire training dataset is used for the UBM, T-Matrix and PLDA training.

B. Corpus 2 and the Baseline Systems

Corpus 2 consists of Switchboard (SWB) training data (LDC97S62) and evaluation data (a subset of LDC2002S09), which was originally collected for speech recognition [30]. Kaldi swbd/s5 is employed for pre-processing such that each conversation is segmented into short utterances [38]. For SWB training set, there are 521 speakers in total, a set of 500 speakers were randomly selected from the SWB training set for triplet network training, while the remaining 21 speakers are used for validation.

For the evaluation, a total of 3234605 trials were created from the evaluation segments by exhausting all possible pairs, the trial list is accessible from our site.[2]

[2]https://github.com/heimanba89/SWB_SV

Similar to the configuration of baseline for Corpus 1, a UBM i-Vector system is developed as one baseline for Corpus 2. The only difference is that LDA is added for dimension reduction (400-D reduced to 300-D) before a PLDA classifier.

It is acknowledged that DNN brings additional SV performance improvement for speech data with transcriptions [7]. In this study, a DNN i-Vector system is also developed based on Kaldi (swbd/s5 & SRE10/v2), where the UBM acoustic model is replaced by a more supervised DNN model. The DNN architecture consists of 6 fully connected hidden layers with 1024 nodes for each layer. A cross-entropy objective function is employed to estimate the posterior probabilities of 3178 senones. An 11-frame context of 39 dimensional ($\Delta + \Delta\Delta$) MFCC feature are projected into 40 dimensions using a fMLLR transform for each utterance [39], which relies on a GMM-HMM decoding alignment. The reason we apply the fMLLR feature here is that, by speaker normalization, it is expected to acquire more accurate phonetic alignment for the following TV matrix training (see more details in [40]). After i-Vector extraction, the same back-ends developed for the UBM i-Vector system are applied.

C. Corpus 3 and the Baseline Systems

Corpus 3 consists of SWB training data and NIST SRE2004, 2005, 2006, 2008 corpus. And we test on NIST SRE10 10 s–10 s condition.

We develop two i-Vector baselines for the SRE10 10 s–10 s evaluation: a UBM i-Vector system and a DNN i-Vector system. For the UBM based model, we extract 60 dimension MFCC features within a 25 ms window, with a shift size of 10 ms. Non-speech frames are discarded using an energy-based VAD. 2048-mixture full covariance UBM and TV Matrix are trained using SRE04-08. At the back-end level, i-Vectors are length normalized and the dimension is reduced from 600 to 400 using LDA. The PLDA classifier is trained with the i-Vectors from SRE04-08 set. The DNN i-Vector model follows almost the same pipeline as the UBM model except for the posterior estimation part, where a SWB ASR acoustic model (developed for Corpus 2) is employed to generate frame-level posterior for TV Matrix training and i-Vector extraction.

V. EXPERIMENTS

In this section, configurations of triplet sampling for different datasets is presented first, followed by a description of input features of the speaker embedding systems. After the summarization of experimental, detailed results are reported as follows.

A. Triplet Sampling for Three Different Datasets

An efficient triplet sampling is important for speaker embedding system training. Table III lists all the triplet sampling configurations in our experiments. The choice of #SPK, #UTT and batch size depends on individual dataset. For example, the #UTT of each speaker of SRE data varies in a large range, while the number total speakers is at 4k level. For this consideration, it is appropriate to use a large #SPK while keep #UTT a relative small number to ensure a sample balanced training for each

TABLE III
TRIPLET SAMPLING FOR ONE EPOCH ON THREE DIFFERENT DATASETS

	#SPK	#UTT	#max_triplet	batch size
Corpus 1	60	40	46800	90
Corpus 2	80	30	34800	60
Corpus 3	240	20	45600	60

#SPK is the speaker number m, #UTT is the segment number per speaker n, #max_triplet is $mn(n-1)/2$, batch size the number of samples (e.g., a batch size 90 contains 90/3 = 30 triplets) for network update.

TABLE IV
A SUMMARIZATION OF FEATURE EXTRACTION FOR THREE DIFFERENT DATASETS

	scale	sample rate	frame length	4s output
Corpus 1	linear	16 kHz	32ms	160×250
Corpus 2	linear	8 kHz	32ms	128×250
Corpus 3	linear	8 kHz	32ms	128×250
Corpus 3	mel	8 kHz	32ms	120×250

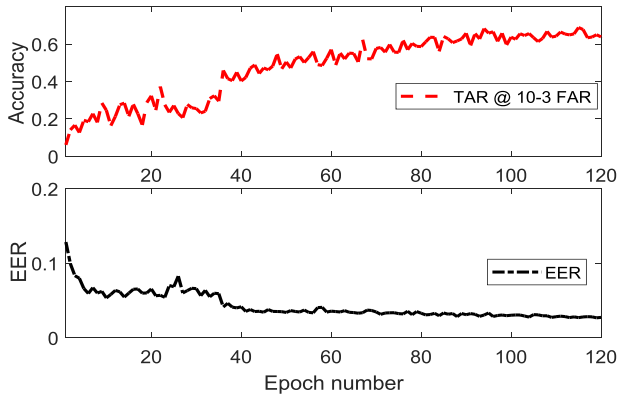

Fig. 7. The SV performance across epochs on Corpus 1. Learning rate =0.1 is used for the first 36 epochs, 0.01 is used until 60 epochs, since than a decay rate 0.5 is applied every 20 epochs, and the RMSProp [41] optimizer is employed throughout the learning process.

speaker. It is noted that the triplet sampling would occasionally find a speaker with the number of samples less than #UTT. That is the reason we note $mn(n-1)/2$ as the #max_triplet, instead of actual number of triplets.

B. Input Features for Speaker Embedding Systems

We evaluate two kinds of features as the input to the Inception-resnet-v1 network: a linear scale spectrogram and a mel-scale fbank feature. The configuration of feature extraction varies according to wavfiles of individual dataset. Table IV summarizes the parameters of different feature extraction methods. For Corpus 1 with the 16 kHz sample-rate, assuming a 0–5K frequency and 4 s time-axis range of interests, the linear scale spectrogram operation results in a 160×250 2-d feature matrix using a 512 point FFT. It is noted that the height and width of the spectrogram will change according to the selected frequency bins and duration. As for 8 kHz datasets (i.e., Corpus 2 and Corpus 3), a 256 point FFT and 0-4K frequency bin in a 4s-segment will produce a feature matrix of dimension 128×250. We also employ a mel-fbank feature for Corpus 3 (NIST SRE10 10 s–10 s condition), a 40-dimension mel-fbank and Δ and $\Delta\Delta$ with a 32 ms frame-length and 50% overlap leads to a 120×250 feature matrix for a 4s-segment.

C. Experiments on Corpus 1

In this section, the speaker embedding system with distance scoring (i.e., named after "e2e" for simplicity throughout Section V-C) on fixed-length input is evaluated with respect to learning rate/epochs, duration conditions, number of enrollment utterances. Next, the fixed-length constraint is removed with the SPP modification and the "incremental" method, the speaker embedding dimension is 128 for all the models developed on Corpus 1. Finally, the advantages of our proposed methods are analyzed over the i-Vector/PLDA system for various duration conditions.

1) Learning Curves on Validation Set: To observe the performance of the proposed fixed 4 s input "e2e" system on the validation set, two different metrics are utilized in the experiment. The first is true acceptance rate (TAR) at a specified false acceptance rate (FAR), written as "TAR@10^{-3}FAR". The definition of TAR and FAR is given as:

$$\mathrm{TAR} = \frac{\#\mathrm{TA}}{\#\mathrm{Target}}, \mathrm{FAR} = \frac{\#\mathrm{FA}}{\#\mathrm{Nontarget}}, \tag{9}$$

where $\#\mathrm{TA}, \#\mathrm{FA}, \#\mathrm{Target}, \#\mathrm{Nontarget}$ are the number of true accepts, false accepts, target trials and nontarget trials respectively. By setting a threshold such that a very low FAR (i.e., 10^{-3}) is fixed, TAR will increase with the training progresses.

Another metric employed is equal error rate (EER). The stop criteria for training is based on these two metrics. In fact, as shown in Fig. 7, TAR still improved even EER saturates after 60 epochs. In the test set, better performance is obtained from the model with a higher TAR, although the EER is almost the same on validation set.

TABLE V
UTTERANCE LEVEL SPEAKER VERIFICATION PERFORMANCE ON TEST SET OF CORPUS 1, RESULTS FROM 3 DIFFERENT TRAINING STAGES (E.G., "E2E/40 E" REPRESENTS THE E2E SYSTEM IS TRAINED WITH 40 EPOCHS) ARE PRESENTED

system	e2e/40 E	e2e/80 E	e2e/120 E	i-Vector/PLDA	fusion
EER	3.98%	**3.26%**	**2.97%**%	3.58%	**2.87%**

2) Performance on Test Set: Table V illustrates results of the "e2e" system on the fixed 4 s condition. The best "e2e" system achieves a $+17.0\%$ relative improvement over the i-Vector/PLDA baseline. An equal weight score fusion of "e2e 120 E" and "i-Vector/PLDA" further boosts performance $+19.8\%$ due to the significant architectural differences between the end-to-end system and the i-Vector baseline.

3) Performance Against Shorter Duration: Fig. 8 shows DET curves of the 4 systems. More specifically, 4 s, 3 s and even shorter 2s conditions are tested with the "e2e" framework. In Fig. 8, a performance degradation is observed when the duration is reduced from 4 s to 3 s or 2 s. In terms of EER, the 3 s condition has a comparable performance with the i-Vector/PLDA system (slightly better: 3.43% VS. 3.58%). It is also interesting to note that three "e2e" systems behave consis-

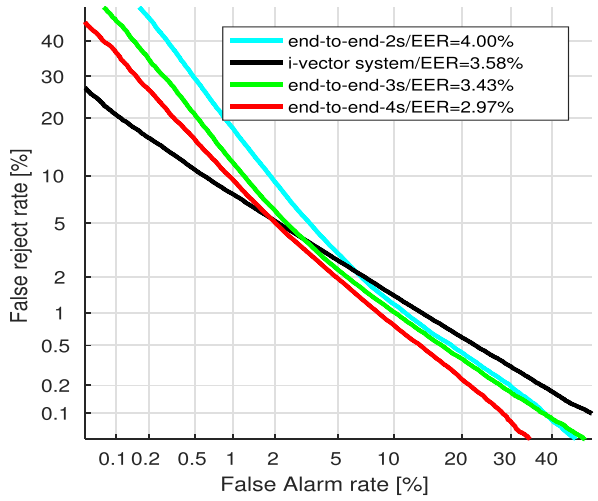

Fig. 8. E2E system DET curvess on different duration conditions of Corpus 1 with comparison with the i-Vector/PLDA system.

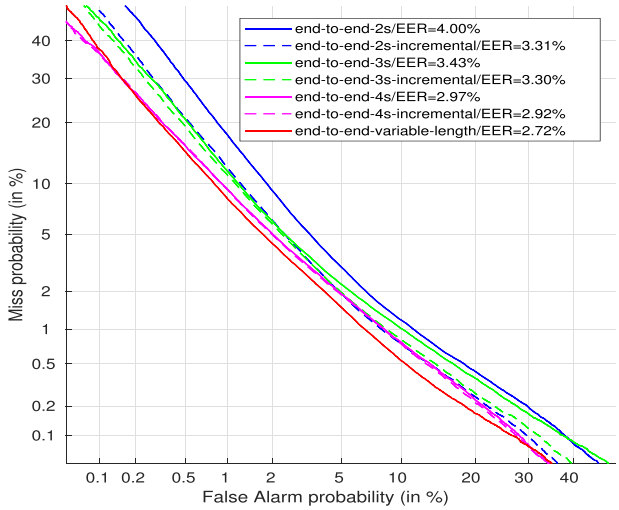

Fig. 9. The DET curves of incremental method and spatial pyramid pooling for variable-length input utterances on Corpus 1.

TABLE VI
SPEAKER LEVEL SPEAKER VERIFICATION PERFORMANCE ON TEST SET, WITH 1, 2, 5, 10 UTTERANCES FOR ENROLLMENT IN TERMS OF EER, ON CORPUS 1

# enroll utts	1	2	5	10
i-Vector/PLDA	3.58%	2.76%	2.03%	1.97%
end-to-end	**2.97%**	**2.41%**	**1.94%**	**1.84%**

TABLE VII
THE EERS OF I-VECTOR/PLDA, END-TO-END 4 S AND END-TO-END VARIABLE LENGTH, ON THE TEST SET OF CORPUS 1

i-Vector/PLDA	e2e 4s	e2e variable length
3.58%	2.97%	2.72%

tently: better performance in the False Alarm dimension while lower performance in the False Reject dimension compared with the i-Vector/PLDA system. This observation suggests that some form of fusion with i-Vector system would be a good option in practice, and potentially to help improve/balance out false accepts/rejects. Besides that, a high false rejection rate indicates that the embeddings from the same speaker are not sufficiently close, which shows that there is still some room to improve the triplet training.

4) Performance Against Number of Enrollment Utterances: It should be noted that both validation and test trial lists are at the utterance level. To evaluate how multiple enrolled utterances influence SV performance, scores are averaged across enrollment utterances to make a final speaker level decision [35].

Results are presented with different number of utterances for enrollment in Table VI. The enrollment action is performed at the score level by averaging scores from the same test utterance. More than 37% of the relative improvement has been achieved when the number of enrolled utterance increases from 1 to 10. From Table VI, a relatively greater improvement is observed when 2 and 5 utterances are enrolled for the speaker model, while the gains with multiple utterance enrollment begins to level off after 5 utterances.

5) Performance for "e2e" System With Spatial Pyramid Pooling: Table VII lists the EERs of three systems: i-Vector/PLDA, "e2e" fixed 4 s (best single system in [22]) and "e2e" variable length with SPP. In terms of EER, the end-to-end variable length with SPP achieves +8.4% relative improvement over "e2e" 4 s system, and +24.0% over the i-Vector/PLDA system. With SPP applied on the Inception-resnet-v1, while the input length constraint is removed, there is SV performance improvement as well.

6) Performance Comparison on Incremental Method and Spatial Pyramid Pooling Method: this section compares two alternative ways in which to handle variable length utterances within the "e2e" framework. As demonstrated in Fig. 9, the "e2e" variable length with the SPP modification gives the best performance, and all "e2e" incremental systems have performance improvement over their respective fixed length versions. It can be seen that improvement is shrinking with better base models, which indicates the limitation of the incremental method.

7) Performance on Different Duration Conditions: A clear performance boost has been shown by removing the fixed length constraint from the "e2e" system. In this experiment, we demonstrate how this "e2e" variable length behaves on different duration conditions. To do so, the test trial list is sorted in terms of test utterance duration, and equally split the list into 6 small lists. The duration ranges are depicted in Fig. 10. For comparison purposes, the results are illustrated from the baseline i-Vector/PLDA and "e2e" 4 s system.

Several observations can be seen from Fig. 10: 1) the i-Vector/PLDA system improves along with the test utterance duration axis; 2) both "e2e" systems have significant advancements over i-Vector/PLDA system for short utterances. 3) "e2e" variable length retains the performance on short durations, while compensating on the long duration portions; 4) the network seems to learn the duration patten as it has the best result for the "3.55–4.5 s" range, where it has the most samples in the training set.

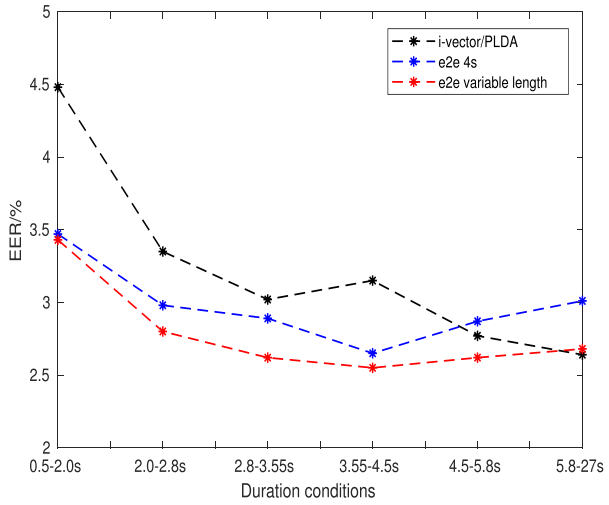

Fig. 10. SV performance (EERs) against different duration conditions on Corpus 1.

D. Experiments on Corpus 2

On Corpus 1, it has been shown that the end-to-end system achieves impressive performance gains compared with a conventional i-Vector/PLDA method. In this section, the performance is evaluated on SWB corpus, where DNN/i-Vector can be implemented for system comparison. In Corpus 2 experiments, the "e2e" system is split into separate parts: the network is trained for speaker embedding extraction, followed by application of standard back-ends such as CDS and PLDA for speaker verification [10], [12], [42]. To ensure the same hyper-parameter for both speaker embedding and i-Vector, we modify the dimension for speaker embedding network to 400-D, same as the i-Vector.

1) Performance on SWB Evaluation Data: as described in Section IV-B, two formulations of i-Vector representations are developed for performance assessment. Results are reported for triplet speaker embedding with variable-length input Inception-resnet-v1, since the best performance was obtained with this architecture on Corpus 1.

Fig. 12 is the DET curves of resulting four systems. As shown in the DET curves, the DNN/i-Vector system outperforms UBM version with a relative +24.9% gain in terms of EER. This is expected based on other studies in the literature [6], [7], [43]. Triplet speaker embedding with CDS back-end produces an EER value which is similar to a DNN/i-Vector system, and a replacement with PLDA achieves the best performance for the SWB corpus.

Besides the overall SV performance, a similar trend was illustrated with Corpus 2, which is consistent with previous experiments: triplet speaker embedding systems have relatively lower performance in the false rejection dimension. This observation indicates a potential weakness of triplet loss based speaker embedding. While there are several ways to normalize intra-speaker variability, for example, train a secondary network with center loss [43] on top of triplet speaker embeddings is a promising idea to address this problem. However, center loss normalization is beyond the scope of this study, and suggested for future work.

2) Visualization of Speaker Representations, i-Vectors and Triplet Speaker Embeddings: A more intuitionistic way to observe separated speakers in the decision space is to project the speaker representations into a 2-D space. To do so, a PCA is performed to reduce the dimensionality to 50 (either 400-D i-Vectors or triplet speaker embeddings). We adopt the same parameter T-SNE training setup for all three different speaker representations. The T-SNE embedding is then applied for this 2-D scatter plot Fig. 11 [45]. While it remains a question as to whether the speaker cluster is fully grouped together, as it does appear to take more space for the triplet speaker embedding cluster than the either UBM or DNN i-Vector clusters. From the scatter plots, the observation that the triplet speaker embeddings are well separated compared with the UBM or DNN i-Vectors, which in turn explains the reason of SV performance gains from our proposed triplet loss based speaker embeddings.

E. Experiments on Corpus 3

Table VIII details the SV performance of proposed speaker embedding systems on the NIST SRE10 10 s–10 s condition. In order to facilitate the system comparisons, the result from [24] is also listed here as a reference. In order to to balance the contradiction between the GPU memory limitation and long dration SRE training data, an utterance segmentation process should be conducted before speaker embedding network training. In practice, a 6 s and a 8 s segmentation is performed on SRE04-08 training data. The detailed speaker embedding system setup is included in Table VIII. The major components (e.g., duration, VAD, feature type, classifier) which are often reported to significantly affect SV performance are examined thoroughly.

One can see from the above results, 8 s segment version speaker embedding system achieves +11.8% relative improvement over SPK_EMB2, which is developed on 6 s speech segments. Compared with other speaker embedding systems, SPK_EMB3 is clearly worse. The result again indicates us that VAD is very important for speaker embedding systems on SRE data. It is also noted here that mel scale features bring additional performance gain, which shows the feature engineering is still important in the deep neural network based frameworks. Finally, we arrive at SPK_EMB5, a speaker embedding + PLDA based SV system which achieves state-of-the-art single system performance for NIST SRE10 10 s–10 s condition.

Compared with our in-house developed baselines, we see that the SPK_EMB5 are +20.3% and +15.9% better against UBM i-Vector and DNN i-Vector respectively. For across group comparison, we have a slightly better performance in the single embedding perspective, while remaining a 12.2% performance gap with their "embeddings" system. As indicated in [24], their "embeddings" are extracted from two different layers of the same network, which could be an interesting future direction to explore the capability of neural network based speaker embedding systems.

VI. Conclusion

In this study, a novel text-independent speaker embedding system for speaker verification was proposed. Triplet loss function allows us to discriminatively train a speaker embedding

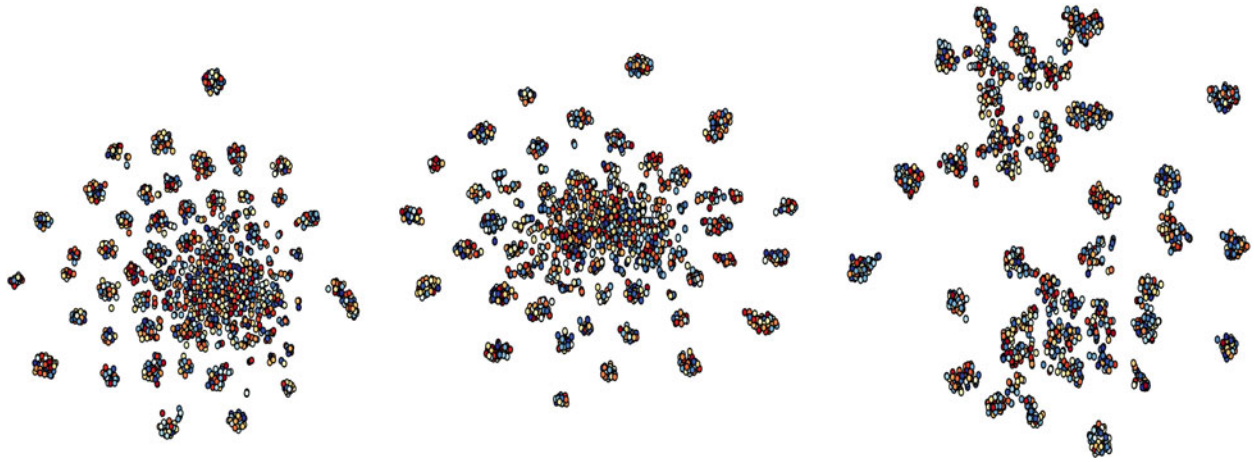

Fig. 11. 2-D PCA+T-SNE scatter plots for UBM/i-Vector, DNN/i-Vector and speaker embedding for test utterance of Corpus 2 (in the order of left, middle and right respectively). It is noted that each cluster represents a speaker, while the colour in this plot does not have meanings.

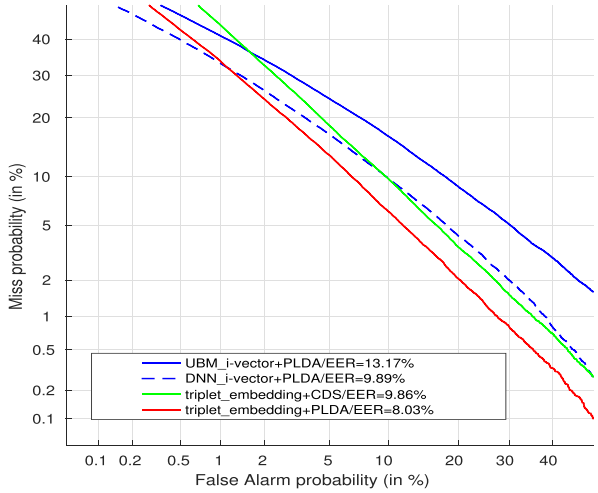

Fig. 12. EERs of i-Vector and triplet embedding systems with CDS or PLDA back-end classifiers on Corpus 2.

TABLE VIII
EER (%) ON SRE10 10 S–10 S CONDITION

System	seg_dur	VAD	feats	scoring	EER %
SPK_EMB1	6s	✓	linear	PLDA	10.7
SPK_EMB2	6s	✓	mel	PLDA	10.2
SPK_EMB3	8s	×	mel	PLDA	14.9
SPK_EMB4	8s	✓	mel	CDS	11.4
SPK_EMB5	8s	✓	mel	PLDA	**9.0**
UBM i-Vector	/	✓	MFCC	PLDA	11.3
DNN i-Vector	/	✓	MFCC	PLDA	10.7
embedding a [25]	/	✓	MFCC	PLDA	11.0
embedding b [25]	/	✓	MFCC	PLDA	9.2
embeddings [25]	/	✓	MFCC	PLDA	**7.9**
i-Vector [25]	/	✓	MFCC	PLDA	11.0

"/" denotes that the system training is not depended on fixed duration segments.

system with deep neural networks. A very deep CNN based network architecture named Inception-resnet-v1 was successfully employed for extracting speaker embeddings. An euclidean distance was integrated in both triplet loss calculation and similarity measure within the trials, which ensures the speaker verification to follow an end-to-end manner. Additionally, we also proved that an L_2-normalized speaker embedding with negative Euclidean distance is equivalent to classical cosine distance scoring solution. This study also focused on duration variability and its influence on speaker verification performance. To relax the constraint of fixed-length-input for the previous framework, two strategies were proposed to 1) "incremental" duration compensation, 2) and replace the final average pooling layer with a Spatial Pyramid Pooling layer within the Inception-resnet-v1 architecture. Both methods improved the overall performance, and the "end-to-end variable length with spatial pyramid pooling" solution achieved the best overall performance. All above components can be attributed to our technical contributions of this study.

Extensive experiments were conducted on three publicly accessible datasets. Compared with the i-Vector/PLDA systems, competitive/better performances were consistently reported across three different datasets with speaker embedding systems with a simple distance scoring method. For Corpus 1, a $+17.0\%$ relative improvement was achieved with a fixed 4 s speaker embedding system with CDS, and a $+24.0\%$ relative improvement with a variable-length input speaker embedding system with CDS. For Corpus 2, The introduction of the PLDA back-end into the triplet speaker embedding system brings additional performance gain over the conventional i-Vector systems and the end-to-end system. Based on this, a $+39.0\%$ and a $+18.8\%$ relative performance gain with respect to a UBM/i-Vector and a state-of-the-art DNN/i-Vector system was achieved. For the challenging NIST SRE10 10 s–10 s, we evaluated different combinations of techniques within the triplet speaker embedding framework which are essential for SV task. By doing this, we hope to share more insights which are able to contribute to this emerging direction for speaker verification research.

It is suggested that this study shows the potential of the neural network based speaker embedding system for speaker recognition and related areas. In essence, this study is to change speaker recognition from a design problem with many separated training modules to a learning problem with neural networks. There are

still many possibilities to improve the speaker embeddings system, for example, employing robust features, applying different neural network architectures, adding alternative loss functions (in our case, adding a center loss like normalization term which is promising to reduce high false rejection rate [44]), etc. At the same time, this approach can be directly applied to many other applications such as speaker change detection, speaker diarization, and speaker adaption for speech recognition [46], [47]. Other directions such as language identification, spoofing detection, stress/emotion recognition from speech are also promising [48]–[50]. The study therefore highlights effective methods for text-independent speaker recognition, as well as fundamental observations for future work.

ACKNOWLDGEMENTS

The authors would like to thank I. Zharkov, U. Batricevic of Microsoft, C. Yu, and other researchers from the Center for Robust Speech Systems, UTDallas for their many insights and helpful discussions during the development of this paper.

REFERENCES

[1] J. P. Campbell, "Speaker recognition: A tutorial," *Proc. IEEE*, vol. 85, no. 9, pp. 1437–1462, Sep. 1997.

[2] J. H. L. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 74–99, Nov. 2015.

[3] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, 2016, pp. 5115–5119.

[4] S. Zhang, Z. Chen, Y. Zhao, J. Li, and Y. Gong, "End-to-end attention based text-dependent speaker verification," in *Proc. IEEE Workshop Spoken Lang. Technol.*, 2016, pp. 171–178.

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

[6] F. Richardson, D. Reynolds, and N. Dehak, "Deep neural network approaches to speaker and language recognition," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1671–1675, Oct. 2015.

[7] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2014, pp. 1695–1699.

[8] S. O. Sadjadi *et al.*, "The 2016 NIST speaker recognition evaluation," in *Proc. INTERSPEECH*, 2017, pp. 1353–1357.

[9] S. Sadjadi, J. Pelecanos, and S. Ganapathy, "The IBM speaker recognition system: Recent advances and error analysis," in *Proc. INTERSPEECH*, 2016, pp. 3633–3637.

[10] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Lang., Process.*, vol. 19, no. 4, pp. 788–798, May 2011.

[11] C. Zhang, F. Bahmaninezhad, S. Ranjan, C. Yu, N. Shokouhi, and J. H. L. Hansen, "UTD-CRSS systems for 2016 NIST speaker recognition evaluation," in *Proc. INTERSPEECH*, 2017, pp. 1343–1347.

[12] P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, and P. Dumouchel, "PLDA for speaker verification with utterances of arbitrary duration," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, 2013, pp. 7649–7653.

[13] T. Hasan, R. Saeidi, J. H. L. Hansen, and D. A. van Leeuwen, "Duration mismatch compensation for i-vector based speaker recognition systems," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2013, pp. 7663–7667.

[14] L. Ferrer, Y. Lei, M. McLaren, and N. Scheffer, "Study of senone-based deep neural network approaches for spoken language recognition," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 24, no. 1, pp. 105–116, 2016.

[15] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digit. Signal Process.*, vol. 10, no. 1, pp. 19–41, 2000.

[16] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Process. Lett.*, vol. 13, no. 5, pp. 308–311, 2006.

[17] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 4, pp. 1435–1447, 2007.

[18] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network based speaker embedding for end-to-end speaker verification," in *IEEE SLT*, 2016.

[19] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, 2017, pp. 4930–4934.

[20] H. Bredin, "TristouNet: Triplet loss for speaker turn embedding," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, 2017, pp. 5430–5434.

[21] C. Zhang, C. Yu, and J. H. L. Hansen, "An investigation of deep-learning frameworks for speaker verification antispoofing," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 4, pp. 684–694, Jun. 2017.

[22] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances," in *Proc. INTERSPEECH*, 2017, pp. 1487–1491.

[23] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[24] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proc. INTERSPEECH*, 2017, pp. 999–1003.

[25] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. INTERSPEECH*, 2017, pp. 2616–2620.

[26] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with flexibility in utterance duration," in *Proc. IEEE Automat. Speech Recognit. Understanding Workshop*, 2017, pp. 584–590.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 346–361.

[28] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proc. Odyssey*, 2010, pp. 14–24.

[29] Y. Lei, L. Burget, L. Ferrer, M. Graciarena, and N. Scheffer, "Towards noise-robust speaker recognition using probabilistic linear discriminant analysis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2012, pp. 4253–4256.

[30] W. Xiong *et al.*, "Achieving human parity in conversational speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, arXiv:1610.05256, 2017.

[31] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. INTERSPEECH*, 2011, pp. 249–252.

[32] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, Inception-Resnet and the impact of residual connections on learning." in *Proc. 31st AAAI Conf. Artifi. Intell.*, 2017, pp. 4278–4284.

[33] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[35] G. Liu and J. H. Hansen, "An investigation into back-end advancements for speaker recognition in multi-session and noisy enrollment scenarios," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1978–1992, Dec. 2014.

[36] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, 2013, pp. 6645–6649.

[37] R. Johnson and T. Zhang, "Effective use of word order for text categorization with convolutional neural networks," in *Proc. Conf. North Amer. Ch. Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2015, pp. 103–112.

[38] D. Povey *et al.*, "The kaldi speech recognition toolkit," in *Proc. IEEE IEEE Workshop Automat. Speech Recognit. Understanding*, 2011.

[39] D. Povey and G. Saon, "Feature and model space speaker adaptation with full covariance Gaussians," in *Proc. INTERSPEECH*, 2006, pp. 1145–1148.

[40] S. O. Sadjadi, S. Ganapathy, and J. Pelecanos, "The IBM 2016 speaker recognition system," in *Proc. Odyssey*, 2016.

[41] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," in *Proc. COURSERA: Neural Netw. Mach. Learn.*, vol. 4, no. 2, pp. 26–31, 2012.
[42] N. Dehak, R. Dehak, J. R. Glass, D. A. Reynolds, and P. Kenny, "Cosine similarity scoring without score normalization techniques," in *Proc. Odyssey*, 2010, pp. 15–19.
[43] C. Yu, C. Zhang, F. Kelly, A. Sangwan, and J. H. Hansen, "Text-available speaker recognition system for forensic applications," in *Proc. INTERSPEECH*, 2016, pp. 1844–1847.
[44] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 499–515.
[45] L. v. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. Nov., pp. 2579–2605, 2008.
[46] H. Dubey, L. Kaushik, A. Sangwan, and J. H. L. Hansen, "A speaker diarization system for studying peer-led team learning groups," in *Proc. INTERSPEECH*, 2016, pp. 2180–2184.
[47] C. Yu, A. Ogawa, M. Delcroix, T. Yoshioka, T. Nakatani, and J. H. L. Hansen, "Robust i-vector extraction for neural network adaptation in noisy environment," in *Proc. INTERSPEECH*, 2015, pp. 2854–2857.
[48] C. Yu *et al.*, "UTD-CRSS system for the NIST 2015 language recognition i-vector machine learning challenge," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, 2016, pp. 5835–5839.
[49] C. Zhang, G. Liu, C. Yu, and J. H. L. Hansen, "I-vector based physical task stress detection with different fusion strategies," in *Proc. INTERSPEECH*, 2015, pp. 2689–2693.
[50] C. Zhang *et al.*, "Joint information from nonlinear and linear features for spoofing detection: An i-vector/DNN based approach," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, 2016, pp. 5035–5039.

Chunlei Zhang received the B.S. degree in environmental engineering and the M.S. degree in acoustics from Northwestern Polytechnical University, Xi'an, China, in 2011 and 2014, respectively. He has been working toward the Ph.D. degree as a Research Assistant in Erik Jonsson School of Engineering and Computer Science, The University of Texas at Dallas, Richardson, TX, USA, since August 2014. His research interests include robust speaker recognition in train/test mismatched conditions, stress/emotion detection, speech recognition, machine learning, and deep learning.

Kazuhito Koishida (M'00) received the B.E. degree in electrical and electronic engineering and the M.E. and Dr.Eng. degrees in intelligence science from Tokyo Institute of Technology, Tokyo, Japan, in 1994, 1995, and 1998, respectively. From 1998 to 2000, he was a Postdoctoral Researcher with the Signal Compression Laboratory, University of California, Santa Barbara, Santa Barbara, CA, USA. He joined the Microsoft Corporation, Redmond, WA, USA, in 2000, where he is currently a Principal Lead Scientist with the Applied Science Group. His research interests include speech and audio processing, multimodal signal processing, and machine learning. He is a member of the ISCA.

John H. L. Hansen (S'81–M'82–SM'93–F'07) received the B.S.E.E. degree from Rutgers University, College of Engineering, New Brunswick, NJ, USA, in 1982, and the M.S. and Ph.D. degrees in electrical engineering from Georgia Institute of Technology, Atlanta, GA, USA, in 1983 and 1988, respectively.

He joined Erik Jonsson School of Engineering and Computer Science, The University of Texas at Dallas (UTDallas), Richardson, TX, USA, in 2005, where he is currently the Jonsson School Associate Dean for Research, as well as a Professor in electrical engineering, and also holds the Distinguished University Chair in telecommunications engineering. He was with the Department Head of Electrical Engineering from August 2005 to December 2012, overseeing a $+4$x increase in research expenditures ($4.5M to $22.3M) with a 20% increase in enrollment and the addition of 18 T/TT faculty, growing UTDallas to be the eighth largest EE program from ASEE rankings in terms of degrees awarded. He also holds a joint appointment as a Professor with the School of Behavioral and Brain Sciences (speech and hearing). At UTDallas, he established the Center for Robust Speech Systems (CRSS), which is part of the Human Language Technology Research Institute. He was the Department Chairman and a Professor with the Department of Speech, Language and Hearing Sciences and a Professor with the Department of Electrical and Computer Engineering, University of Colorado Boulder, Boulder, CO, USA (1998–2005), where he cofounded and was the Associate Director of the Center for Spoken Language Research. In 1988, he established the Robust Speech Processing Laboratory and continued to direct research activities in CRSS, UTDallas. He has supervised 85 Ph.D./M.S. thesis candidates (36 Ph.D. and 37 M.S./M.A. candidates), was recipient of The 2005 University of Colorado Teacher Recognition Award as voted on by the student body, author/co-author of 685 journal and conference papers including 12 textbooks in the fields of machine learning based speech processing and language technology, and signal processing for vehicle systems. He is the coauthor of numerous speech/language textbooks and lead author of The Impact of Speech Under Stress on Military Speech Technology (NATO RTO-TR-10, 2000). He has been named an IEEE Fellow (2007) for contributions in "Robust Speech Recognition in Stress and Noise" and an International Speech Communication Association (ISCA) Fellow (2010) for contributions on research for speech processing of signals under adverse conditions. He currently serves as President of ISCA and a member of the ISCA Board. He was also selected, and is serving, on the U.S. Office of Scientific Advisory Committees (OSAC) for OSAC-Speaker in the voice forensics domain (2015–2017). He was an IEEE Technical Committee Chair and Member of the IEEE Signal Processing Society: Speech-Language Processing Technical Committee (SLTC) (2005–2008 and 2010–2014; elected the IEEE SLTC Chairman for 2011–2013 and the Past-Chair for 2014), and elected an ISCA Distinguished Lecturer (2011/12). He was also a member of the IEEE Signal Processing Society Educational Technical Committee (2005–2008 and 2008–2010). He was the Technical Advisor to the U.S. Delegate for NATO (IST/TG-01), an IEEE Signal Processing Society Distinguished Lecturer (2005/06), an Associate Editor for the IEEE TRANSANCTIONS ON SPEECH AND AUDIO PROCESSING (1992–1999), an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS (1998–2000), an Editorial Board Member for the IEEE SIGNAL PROCESSING MAGAZINE (2001–2003). He was a Guest Editor of the October 1994 special issue on robust speech recognition for the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING. He was on the Speech Communications Technical Committee for the Acoustical Society of America (2000–2003), and is currently a member of the ISCA Advisory Council. He also organized and was the General Chair for the ISCA Interspeech 2002, September 16–20, 2002, and a Co-organizer and the Technical Program Chair for the IEEE International Conference on Acoustics, Speech and Signal Processing 2010, Dallas, TX, USA. He was also the Co-Chair and Organizer for the IEEE Spoken Language Technology Workshop 2014, December 7–10, 2014, in Lake Tahoe, NV, USA. He was the recipient of the 2005 University of Colorado Teacher Recognition Award as voted on by the student body, and the Acoustical Society of Americas 25 Year Award (2010) in recognition of his service, contributions, and membership to the Acoustical Society of America.