Speech Communication 99 (2018) 101–113

Contents lists available at ScienceDirect

Speech Communication

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

Front-end speech enhancement for commercial speaker verification systems

Sefik Emre Eskimez*[,a], Peter Soufleris[b], Zhiyao Duan[a], Wendi Heinzelman[a]

[a] University of Rochester, 500 Wilson Blvd, Rochester, NY 14627, USA
[b] Voice Biometrics Group, 12 Penns Trail, Newtown, PA 18966, USA

ARTICLE INFO

Keywords:
Speech enhancement
Automatic speaker verification
Deep neural networks
Bidirectional LSTM
Convolutional encoder-decoder

ABSTRACT

Commercial speaker verification systems are an important component in security services for various domains, such as law enforcement, government, and finance. These systems are sensitive to noise present in the input signal, which leads to inaccurate verification results and hence security breaches. Traditional speech enhancement (SE) methods have been employed to improve the performance of speaker verification systems. However, to the best of our knowledge, the impact of state-of-the-art speech enhancement techniques has not been analyzed for text-independent automatic speaker verification (ASV) systems using real-world utterances. In this work, our contribution is twofold. First, we propose two deep neural network (DNN) architectures for SE, and we compare the performance of the proposed networks with the existing work. We evaluate the resulting SE networks using the objective measures of perceptual evaluation of speech quality (PESQ) and short-time objective intelligibility (STOI). Second, we analyze the performance of ASV systems when SE methods are used as front-end processing to remove the non-stationary background noise. We compare the resulting equal error rate (EER) using our DNN based SE approaches, as well as existing SE approaches, with real customer data and the freely available *RedDots* dataset. Our results show that our DNN based SE approaches provide benefits for speaker verification performance.

1. Introduction

Automatic speaker verification (ASV) systems are vital for security applications in areas such as financial services, law enforcement, and government security. A security breach occurs when an ASV system makes a false authorization for an imposter, which may lead to economic, personal or national security consequences. Noise, reverberation and channel distortion are factors that significantly impair the performance of ASV systems and make the ASV system particularly vulnerable to imposter attacks or missed verification.

Therefore, speech enhancement (SE), which aims to reduce noise in the speech signal, is an important pre-processing module in commercial ASV systems. These systems in general use traditional SE techniques (Boll, 1979; Ephraim and Malah, 1984; 1985), which have been shown to be effective against stationary noise. However, as most noise types encountered in real-world applications are non-stationary, traditional SE techniques do not perform well in these cases.

Deep neural networks (DNNs) have been successfully applied to SE systems to model non-stationary noise (Lu et al., 2012; 2013; 2014; Xu et al., 2014b; 2014a; 2014c; 2015; Huang et al., 2015; Chen et al., 2016; Weninger et al., 2015). However, these techniques have typically been tested in laboratory settings using an artificially created speech corpus (e.g., TIMIT Acoustic-Phonetic Continuous Speech Corpus sentences Garofolo et al., 1993), where the utterances are spoken in a very different way, i.e., not natural, compared to real-world speech utterances. To be able to assess the feasibility of SE systems in commercial applications, these methods need to be evaluated with real-world utterances in addition to artificial tests.

In this work, we propose two DNN-based speech enhancement approaches. We apply them as a front-end noise removal module for a state-of-the-art speaker verification system and test the combined systems. In addition to evaluating the proposed systems using utterances collected and mixed in laboratory settings, we also use utterances that are collected by a commercial ASV system from real customers, as well as the freely available Reddots dataset to evaluate the proposed systems. We show that both systems yield superior results compared to traditional methods, in terms of both objective speech quality and intelligibility measures and speaker verification performance.

* Corresponding author.
E-mail addresses: eeskimez@ur.rochester.edu (S.E. Eskimez), peter.soufleris@voicebiogroup.com (P. Soufleris), zhiyao.duan@rochester.edu (Z. Duan), wendi.heinzelman@rochester.edu (W. Heinzelman).

https://doi.org/10.1016/j.specom.2018.03.008
Received 30 June 2017; Received in revised form 26 February 2018; Accepted 15 March 2018
Available online 16 March 2018
0167-6393/ © 2018 Elsevier B.V. All rights reserved.

2. Related work

In this section, we review existing work on speech enhancement and its application to speaker verification systems.

2.1. Speech enhancement: Classical methods

Early notable works on speech enhancement modeled the noise statistically, typically using the first 4–5 frames of the noisy speech signal, assuming those are noise only. These methods, such as spectral subtraction (SS) (Boll, 1979), minimum mean square error spectral amplitude estimator (MMSE) (Ephraim and Malah, 1984) and minimum mean square error log-spectral amplitude estimator (Log-MMSE) (Ephraim and Malah, 1985), produce disturbing musical artifacts, which are portions of spectral power appearing in random frequency regions, in the predicted signal. Since these techniques use the first frames to model the noise, they are not effective against time-varying noises.

2.2. Speech enhancement: Deep learning methods

In recent years, DNN based methods have been shown to significantly outperform classical methods. Various deep models have been proposed, but generally they can be classified into two categories: *regression-based* and *masking-based*.

Regression-based methods attempt to learn the mapping from noisy speech to clean speech directly. Lu et al. (2012) trained a deep auto-encoder (DAE) on Mel-scale power spectral patches of clean speech and used this to denoise noisy speech. Later, they extended the model by training the DAE with noisy-clean speech pairs (Lu et al., 2013) and by introducing ensemble models (Lu et al., 2014).

Similarly, Xu et al. (2014b, 2014a, 2014c, 2015) used restricted Boltzmann machines (RBMs) to learn a mapping function from the log power spectra of noisy speech to those of clean speech. They extended this work by adding a statistical estimate of the noise from the first several frames to the network's input to achieve noise-aware training (Xu et al., 2014a). In (Xu et al., 2014c), they further extended this work by introducing global variance equalization to tackle the over-smoothing issue that causes the removal of speech segments in the predicted speech, which leads to muffled speech.

Park and Lee (2016) proposed a redundant convolutional encoder-decoder (R-CED) network, which is a fully convolutional network, for mapping the noisy STFT magnitude to clean STFT magnitude. They applied 1D convolution along the frequency axis. The input to the network is eight frames including the current and the past seven frames, where the output is the current frame's clean version.

Masking-based methods, on the other hand, attempt to predict the time-frequency (T-F) filters or masks that are later applied to noisy speech spectra to recover the corresponding clean speech spectra. Methods in this category have shown significant improvements over regression-based methods (Li and Wang, 2009; Erdogan et al., 2015; Narayanan and Wang, 2013; Huang et al., 2015; Wang et al., 2014; Williamson et al., 2016). Various types of masks have been proposed. Binary masks such as the ideal binary mask (IBM) (Srinivasan et al., 2006; Li and Wang, 2009) and the target binary mask (TBM) (Kjems et al., 2009) set the mask value at a T-F unit to 1 when speech dominates and to 0 when noise dominates. Soft masks such as the ideal ratio mask (IRM) (Erdogan et al., 2015) and the Wiener-like mask (Srinivasan et al., 2006; Narayanan and Wang, 2013; Erdogan et al., 2015) use a real value between 0 and 1 to reflect the relative dominance of speech in each T-F unit. An extension to soft masks is a complex soft mask such as the complex ideal ratio mask (Williamson et al., 2016). This mask uses complex numbers and is applied to the complex spectra of the noisy speech. Wang et al. (2014) investigated some of the above-mentioned masks in a supervised simultaneous speech separation system.

Different types of DNNs have been proposed to predict these masks from noisy speech for SE. Chen et al. (2016) trained a feed-forward DNN to predict the IRM from 64-band cochleagrams of the noisy speech. The network was trained with 10,000 different types of noise to increase the robustness against unseen noises. Weninger et al. (2015) used a long short-term memory (LSTM) network to predict *phase sensitive masks*, and tested the use of this speech enhancement system on the performance of a speech recognition system. Huang et al. (2015) proposed a recurrent neural network to jointly output the clean speech, noise, and the IRM. The training objective function considers both the interference reduction and mask prediction.

2.3. Automatic speaker verification

The Gaussian mixture model (GMM) - universal background model (UBM) ASV system described in (Reynolds et al., 2000; Bimbot et al., 2004) utilizes GMMs to model the acoustic space, which is parametrized by the selected acoustic features. A GMM with a typically large number of mixtures is trained using a large pool of speakers. This model is usually called the UBM.

Dehak et al. (2011) proposed a *total variability space* that represents the speaker and channel variability. The speaker's supervector can be represented in the total variability space by the following equation,

$$s = m + Tw, \tag{1}$$

where s is the speaker's supervector, m is the mean supervector of the GMM-UBM, T is the total variability matrix, and w is the latent variable where the maximum a posteriori (MAP) point estimate of w given the utterance is ϕ, which is called the identity vector (i-vector). For the process of training the T matrix and extracting the i-vectors, please see (Kenny et al., 2005) and (Dehak et al., 2011), respectively.

Probabilistic linear discriminant analysis (PLDA) assumes that the i-vector ϕ can be represented by the following equation,

$$\phi_{l,r} = \mu + Fh_l + Gv_{l,r} + \epsilon_{l,r}, \tag{2}$$

where F and G matrices represent the speaker and channel subspace, l and r represent the speaker and session indexes, h_l and $v_{l,r}$ represent the speaker- and session-specific vectors, μ represents the mean i-vector and $\epsilon_{l,r} \sim \mathcal{N}(0, \Sigma)$ represents the residual noise. The PLDA parameters $\theta_{PLDA} = \{\mu, F, G, \Sigma\}$ can be estimated by expectation maximization (EM). The probabilistic form of Eq. (3) is as follows

$$p(\phi_{l,r}) = \mathcal{N}(\phi_{l,r}|\mu, FF^T + GG^T + \Sigma). \tag{3}$$

For detailed information on how to estimate PLDA parameters with EM, how to calculate multi-session PLDA scoring and how to apply length normalization, please refer to (Prince and Elder, 2007; Jiang et al., 2012), (Lee et al., 2013) and (Garcia-Romero and Espy-Wilson, 2011), respectively.

2.4. SE application to ASV systems

Godin et al. (2013) evaluated speaker identification (SID) methods and SID performance improvements using the early (classical) speech enhancement techniques described in Section 2.1 (Boll, 1979; Ephraim and Malah, 1984; 1985) to see if SE is useful in real noisy telephone conversations. They compared the equal error rate (EER) values between artificially generated noisy speech (i.e., adding noise to clean speech) and natural noisy speech, and found that they do not correlate well.

In recent years, deep-learning based speech enhancement methods have also been integrated into ASV and SID systems. Zhao et al. (2011, 2014) proposed a robust SID system under noisy and reverberant conditions where the IBM prediction was adopted for speech enhancement. They integrated SE and SID systems at the feature level.

Kolbœk et al. (2016) proposed an LSTM-based SE front end for a text-dependent i-vector-based ASV system. This SE network includes

two LSTM layers and a fully connected layer. For each audio frame (32 ms window with 16 ms hop size), the input to their network is a concatenation of the magnitude spectra of the current frame and its previous 15 and future 15 frames, totaling 31 frames of data. The output of the network is the T-F mask of the current frame. They trained and evaluated their system using six types of non-stationary noises and compared their results with classical SE methods. They showed that their method outperforms classical methods in an SNR range from -5 dB to 10 dB.

Although this was a good evaluation of SE systems as a denoising front-end, this evaluation had two limitations. First, all noise types that were used for evaluation were used for training; a more thorough analysis using unseen noise types would be required. Second, all noisy speech utterances were created by artificially mixing clean speech utterances with noise; while this made it possible to create noisy speech with different SNRs, an additional evaluation with natural noisy speech would be required to show the SE front end's performance with commercial ASV systems in real-world scenarios.

To the best of our knowledge, there has not been a thorough analysis of state-of-the-art speech enhancement approaches working with commercial text-independent ASV systems in real-world scenarios. As in Kolbœk et al. (2016), we treat the ASV system as a black box: we enhance the noisy speech and then feed it to the ASV system for speaker verification. In our experiments, we use natural noisy speech samples that were collected by Voice Biometrics Group (VBG) and utterances from the *RedDots* dataset and evaluate the verification error rate on these enhanced utterances. In addition, we conduct artificial tests by mixing additional noise to natural noisy speech utterances with different SNRs and evaluate the verification error rate.

3. Network architecture

In this section, we propose two neural network architectures for speech enhancement as the front end of our ASV systems.

3.1. Bidirectional LSTM network

The first architecture we propose has a total of five layers including the input layer, as shown in Fig. 1. Each hidden BLSTM layer contains 1024 units. The input layer receives a sequence of L vectors, each of which corresponds to one time frame of the input noisy speech.

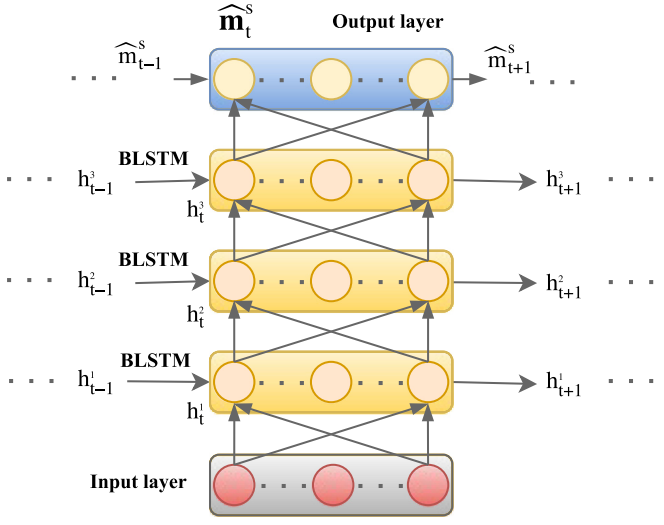

Fig. 1. Proposed BLSTM network architecture for speech enhancement. Input vector $\mathbf{v}_t$ is the concatenation of the normalized log-amplitude spectra of $2c + 1$ frames centered around the t-th time frame, where c is the short-term context window parameter. Hidden layer outputs are denoted as $\mathbf{h}_t^n$, where n is the layer index. $\widehat{\mathbf{m}}_t^s$ is the predicted mask for the speech.

Specifically, each vector is the concatenation of the log-amplitude spectrogram of the $2c + 1$ neighboring frames centered around the current frame, where c is the short-term context window parameter. Including the neighboring frames provides subsequent layers with contextual information. The input then goes through three Bidirectional LSTM (BLSTM) (Hochreiter and Schmidhuber, 1997) layers that model the temporal dependencies of the signal. The output layer consists of a BLSTM layer to reconstruct the speech mask.

We use dropout layers with a 0.2 dropout rate between the BLSTM hidden layers and add l_2 regularization to the network weights during the optimization to overcome overfitting and to increase robustness against unseen noise types. The sigmoid activation function is used in the BLSTM hidden layers.

The BLSTM network is a fully recurrent network, i.e., it only contains BLSTM layers, even in the output. The main difference between the RNN-based method in (Huang et al., 2015) and our network is that we use BLSTM layers instead of basic recurrent layers. Compared to general RNNs, LSTM units are better at modeling long-term temporal dependencies of data, as it suffers less from the vanishing gradient issue (Hochreiter and Schmidhuber, 1997). Our network directly predicts the T-F masks rather than computing it in a deterministic layer as in (Huang et al., 2015; Weninger et al., 2015).

3.2. Convolutional encoder-decoder network

The second network architecture that we propose here is a convolutional encoder-decoder (CED) network, as shown in Fig. 2. The input layer receives a short-time Fourier transform (STFT) magnitude spectrogram of the noisy speech. This input is then passed to four convolutional layers with a stride length of two forming an encoder, followed by three deconvolutional layers (Zeiler et al., 2010) with a stride length of two forming a decoder. This encoder-decoder design compresses and reconstructs the input, and preserves compact and important features. Three skip connections, as denoted by red arrows in the figure, are also added, to help preserve the fine details for better decoding. Finally, a mask for speech is estimated at the output layer. Each of the convolutional and deconvolutional layers also includes a batch normalization (BN) layer and an activation layer with rectified linear unit (ReLU), that are not shown in the figure. The numbers of filters used in all of the convolutional and deconvolutional layers are 128, 256, 512, 1024, 512, 256, 128, and 1, respectively. Filter sizes are 7×7 for all layers, except for the output layers, where filter sizes are 3×3. We add l_2 regularization to the network weights during the optimization to overcome over-fitting and to increase robustness against unseen noise types.

This architecture is inspired by Park and Lee (2016) and Vincent et al. (2010). The main difference between redundant convolutional encoder-decoder (*R-CED*) proposed in (Park and Lee, 2016) and our approach is that we model both speech and noise where R-CED only models the speech. Another difference is that instead of using only 8 STFT frames to denoise a single frame, our network takes much more ($L = 100$ in the experiments) frames and returns the same amount of mask frames. We divide each test utterance into non-overlapping segments that are L frames long and feed each segment into the CED network for enhancement. The rationale behind selecting this much larger number of frames to analyze is that it leads to modeling longer-term dependencies and yielding a better reconstruction. In addition, the network depth is also different, R-CED contains 15 layers and is deeper than CED, where each layer contains a convolution, BN and an activation layer, and the proposed CED has 7 layers, where each layer contains three layers, namely a convolution/deconvolution, a BN layer and an activation layer. The number of filters are symmetric in R-CED blocks which are 10, 12, 14, 15, 19, 21, 23, 25, 23, 21, 19, 15, 14, 12, 10, and 1, while the number of filters in the proposed CED are fixed.

3.3. Objective function

We consider the amplitude soft mask (ASM) in our experiments. ASM for the speech source is defined as

$$\mathbf{m}_t^s(f) = \frac{\mathbf{s}_t(f)}{\mathbf{s}_t(f) + \mathbf{n}_t(f)}, \tag{4}$$

where $\mathbf{s}_t$ and $\mathbf{n}_t$ are the clean speech and the noise magnitude spectra at time t, respectively.

To train the networks, we consider two loss functions, the mean-squared error (MSE) and binary cross-entropy (BCE). The MSE objective function minimizes the reconstruction error of the T-F mask of the speech source of the training data as

$$J_{MSE} = \sum_{t,f} \|\mathbf{m}_t^s(f) - \widehat{\mathbf{m}}_t^s(f)\|^2, \tag{5}$$

where $\mathbf{m}_t^s$ is the mask calculated from the clean speech and the noise, and $\widehat{\mathbf{m}}_t^s$ is the mask that is predicted by the network.

The ground-truth ASM speech mask, whose values range from 0 to 1, can be considered as probabilities of T-F bins belonging to the speech source. The predicted speech mask, whose values also range from 0 to 1, thanks to the sigmoid transfer function at the output layer, can be viewed as the predicted probabilities of T-F bins belonging to the speech source. Therefore, BCE can be used to measure the mismatch between the two Bernoulli distributions as

$$\begin{aligned} J_{BCE} &= \sum_{t,f} H(\mathbf{m}_t^s(f), \widehat{\mathbf{m}}_t^s(f)) \\ &= -\sum_{t,f} \mathbf{m}_t^s(f)\log\widehat{\mathbf{m}}_t^s(f)) + (1 - \mathbf{m}_t^s(f))\log(1 - \widehat{\mathbf{m}}_t^s(f))). \end{aligned} \tag{6}$$

We compare the MSE and BCE objective functions and analyze their effects on speech enhancement performance in Section 4.3.3.

4. Experiments

We divide the experiment section into two parts. The first part evaluates the speech quality and intelligibility of the speech enhancement approaches on noisy speech utterances that are artificially mixed from clean speech and noise. The clean utterances are not naturally encountered by commercial ASV systems and the mixing process is artificial, however, they are needed for calculating the evaluation measures and are publicly available for results reproduction. The second part connects the proposed approaches with a speaker verification system and evaluates their verification error rates on real-world speech utterances.

For training, we create noisy speech sentences by mixing clean speech utterances from the Librispeech corpus (Panayotov et al., 2015) with 138 different types of non-stationary noise obtained from Sound Ideas (Sound, 2018), with SNRs at −6, −3, 0, 3, 6, and 9 dB, totaling about 80 hours of training data. The noise data includes non-stationary noise from various environments such as nature, city, domestic, office, traffic and industry, all of which are what commercial ASV systems may encounter. All files are downsampled to 8 kHz to simulate the telephone frequency range, since many commercial ASV systems use this range. Our proposed networks described in Section 3, namely *BLSTM* and *CED*, are trained once and used in all of the experiments described in this section.

4.1. Comparison methods

As a comparison to our approaches, we trained the fully convolutional redundant CED (*R-CED*) network, described in (Park and Lee, 2016) and in Sections 2.2 and 3.2, as our convolutional baseline .

We designed another DNN-based baseline identical to our BLSTM architecture, but instead of BLSTM layers it uses general recurrent layers, similar to the approach in (Huang et al., 2015). The differences are that we directly predict the masks instead of using a deterministic layer to compute them, and we do not include signal interference terms in the objective function as described in (Huang et al., 2015). We call this network recurrent neural network (*RNN*) for simplicity.

We also compare with traditional SE methods described in Section 2.1, namely SS and Log-MMSE methods. We use implementations provided in (Loizou, 2013).

We implement all DNN-based methods (including the proposed ones) using Keras, a Python library for deep learning (Chollet et al., 2015).

4.2. Speech quality and intelligibility evaluation

We mix 300 utterances of 85 unique speakers with 5 types of noise (babble, factory, speech-shaped noise (SSN), motorcycle and cafeteria) at SNRs of −6, 0, 6 and 9 dB. All of the 85 speakers and the 5 types of noise have not been used as part of the training data. Specifically, the babble and factory noises are obtained from Varga and Steeneken (1993), motorcycle noise is obtained from Duan et al. (2012), the cafeteria noise is recorded by ourselves at the University of Rochester, and the SSN noise is created by filtering white noise with an FIR filter with frequency response that matched the long-term spectrum of speech utterances (Nilsson et al., 1994). We provide the mentioned test noise samples on our website.[1] Fig. 3 shows an example noisy spectrogram corrupted by motorcycle noise at 0 dB SNR along with its corresponding clean and enhanced versions. Among the 300 utterances, 120 are from the Librispeech corpus spoken by 65 unique speakers, and 180 utterances are from the PTDB-TUG corpus (Pirker et al., 2011) spoken by 20 unique speakers. In particular, the inclusion of the PTDB-TUG utterances is to further test the cross-corpora performance of the proposed approach. We use the perceptual evaluation of speech quality (PESQ) (Rix et al., 2001) and short-time objective intelligibility (STOI) (Taal et al., 2011) to evaluate our approaches. Both metrics are widely used in SE research. We do not conduct subjective listening tests, as our primary goal in this work is to analyze the effect of DNN-based SE systems on the performance of an ASV system.

For pre-processing, we perform STFT with a 32 ms Hanning window and an 8 ms hop size to obtain the log-amplitude spectrogram of the noisy speech to be input to all networks. We set FFT size to 256 in our experiments and we use the full frequency range of 0 to 4000 Hz. These parameters are kept the same for all of the speech enhancement experiments. We normalize the input to have zero mean and unit standard deviation. For the BLSTM network, we set the short-term context window parameter c to 5 frames in all experiments. Increasing this parameter yields faster convergence, but at the cost of computational complexity. We empirically set the time sequence length parameter L to 100 frames for both networks. Training the networks on the long input sequences makes the networks more robust to non-stationary noise, which varies over time.

For training, the dropout rate is set to 0.2 for the BLSTM network, and the l_2 regularization value is set to 0.000001 for both networks. The models are trained for 100 epochs, i.e., we iterate over the training set for 100 times. For testing, the network reconstructs the masks of both speech and noise. We then apply the predicted speech mask to the noisy signal's magnitude spectrogram and then reconstruct its time-domain signal using an inverse STFT with overlap-add from the resulting magnitude spectrogram with the noisy speech's phase. We trained both networks using only the BCE objective function described in Eq. (6), as we found that BCE consistently outperforms MSE in our system analysis experiments in Section 4.3.3.

Figs. 4–6 show the PESQ and STOI results for the unprocessed noisy speech and the enhanced speech using the traditional techniques of

[1] http://www.ece.rochester.edu/projects/wcng/code.html.

spectral subtraction (*SS*) and minimum mean square error log-spectral amplitude estimator (*Log-MMSE*) as well as the DNN-based *RNN, R-CED*, and the two proposed networks described in Section 3, namely *BLSTM* and *CED*.

The results show that the proposed techniques (*BLSTM* and *CED*) are superior than other techniques in terms of the PESQ and STOI metrics in completely unmatched noise types and speaker scenarios. *BLSTM* achieves the best improvement in terms of PESQ and STOI, while *CED* achieves the second best results. *SS* and *Log-MMSE* make the STOI values worse than for the unprocessed noisy speech. We believe that this is due to the musical artifacts introduced by the spectral subtraction operation: the amount of subtraction is determined by the estimated instantaneous SNR, but the estimation does not consider long-term temporal dependencies and leads to fluctuating and inappropriate estimation. This issue also leads to degraded performance of the following ASV system, as shown in Section 4.4.2.

4.3. Parameter analysis of the proposed methods

In this section, we further analyze the effects of several key parameters of the proposed CED and BLSTM networks, including the number of hidden units and layers, the objective function, and the input features. In the following experiments, we use the same settings described in Section 4.2, i.e., the train and test speech and noise combinations are the same. We report the average results of five test noise types.

4.3.1. The number of hidden units

We analyze the effect of different numbers of hidden units in the BLSTM network on PESQ and STOI results. We investigate a three-layer BLSTM network with N units in each layer, where N is varied to take values of 64, 128, 256, 512 and 1024. PESQ and STOI results are shown in Fig. 7. The results suggest that increasing the number of hidden units monotonically improves PESQ and STOI across all SNR conditions, yet the improvement seems to be close to saturation when N is 1024. Increasing N beyond 1024 is not feasible for us due to insufficient memory; we used an NVIDIA Tesla K80 GPU which has 12GB memory.

Next, we investigate the effect of different numbers of filters of the CED network on PESQ and STOI results. The CED network has a symmetric encoder-decoder structure, and the number of filters can be described as M, $2M$, $4M$, $8M$, $4M$, $2M$, M for the hidden layers and 1 filter for the predicted speech mask. We vary M to have values of 8, 16, 32, 64 and 128 and show PESQ and STOI results in Fig. 8. Again, we can see that increasing M generally improves PESQ and STOI across all SNR conditions, yet the improvement is very small when M is greater than 32. Increasing M above 128 is not feasible for us due to insufficient memory.

In practice, the trade-off between system performance and computational cost needs to be balanced. In our experiments, we chose N and M to be 1024 and 128, respectively, to achieve the best possible PESQ and STOI on our device.

4.3.2. The number of hidden layers

We investigate the effect of the number of layers in BLSTM and CED networks. For the BLSTM network, we let each hidden layer contain 1024 units and vary the number of hidden layers between 1 and 3. The PESQ and STOI results are shown in Fig. 9. We can see that increasing the number of hidden layers improves both PESQ and STOI across all SNR conditions. Increasing the number of layers above three is not feasible due to insufficient memory.

The CED network has two parts, the encoder and the decoder. In Fig. 2, there are a total of 7 layers shown. We vary this number to 3, 5 and 7 and compare their PESQ and STOI performance. The number of filters of the hidden layers follows the same power of 2 ratio as described in the previous subsection, and we set M to 128. Also note that the number of skip connections also varies to be 1, 2 and 3 for networks with 3, 5 and 7 layers, respectively. Results are shown in Fig. 10. Again, we see that more layers leads to better PESQ and STOI performance across all SNR conditions. However, the number of parameters also increase dramatically, by approximately 11 times from 3 layers to 7 layers.

In our experiments, we set the number of hidden layers to 3 and 7 for the BLSTM and the CED networks, respectively, in order to achieve the best possible PESQ and STOI on our device. Considering the hidden layer size parameters in the previous subsection, the BLSTM and the CED networks have 54,782,992 and 17,669,889 trainable parameters, respectively.

4.3.3. The objective function

This section compares the mean-squared error (MSE) objective function from Eq. (5) and the binary cross-entropy (BCE) objective function from Eq. (6) for CED and BLSTM networks. The results are shown in Fig. 11. We can see that the BCE objective function achieves slight but consistent improvement over the MSE objective function on both metrics and networks and across all SNR conditions. Therefore, we use the BCE objective function in all the remaining experiments.

4.3.4. The input feature

Next, we compare the log-amplitude linear-frequency (log-linear) spectrogram with the log-amplitude mel-frequency (log-mel)

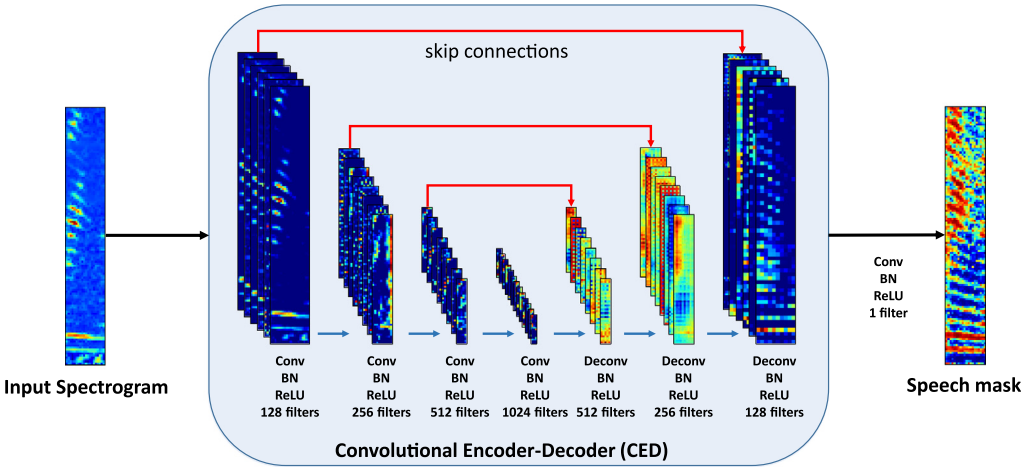

Fig. 2. Proposed convolutional encoder-decoder (CED) network architecture for speech enhancement. The numbers of filters in the convolution and deconvolution layers are 128, 256, 512, 1024, 512, 256, and 128, respectively. The input is an L-frame magnitude spectrogram, where the output are estimated L-frame mask of speech and noise spectrograms. The red arrows represent the skip connections. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

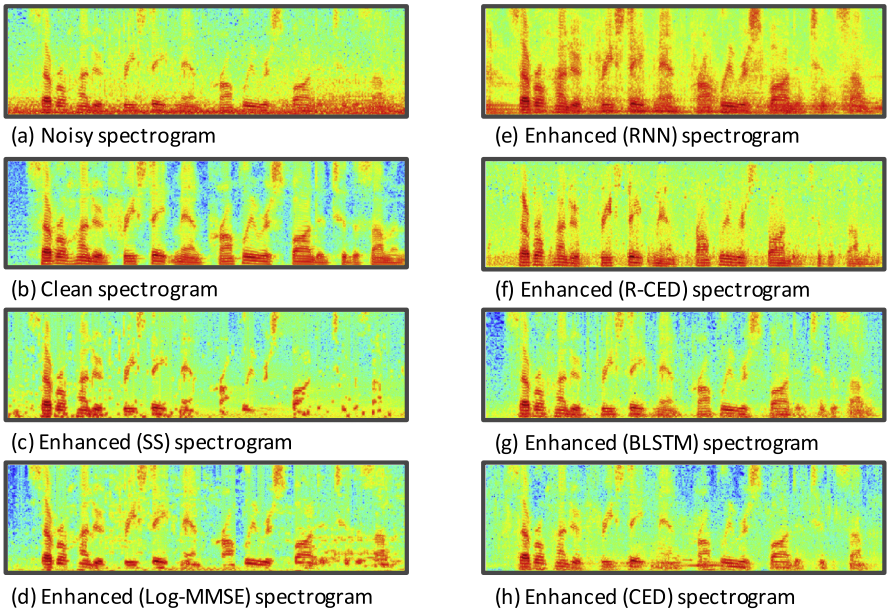

Fig. 3. An example of speech enhancement results. Magnitude spectrograms of the noisy speech signal corrupted by motorcycle noise at 0 dB, the ground-truth clean speech, and enhanced speech of six speech enhancement methods, namely *SS, Log-MMSE, RNN, R-CED, BLSTM* and *CED*.

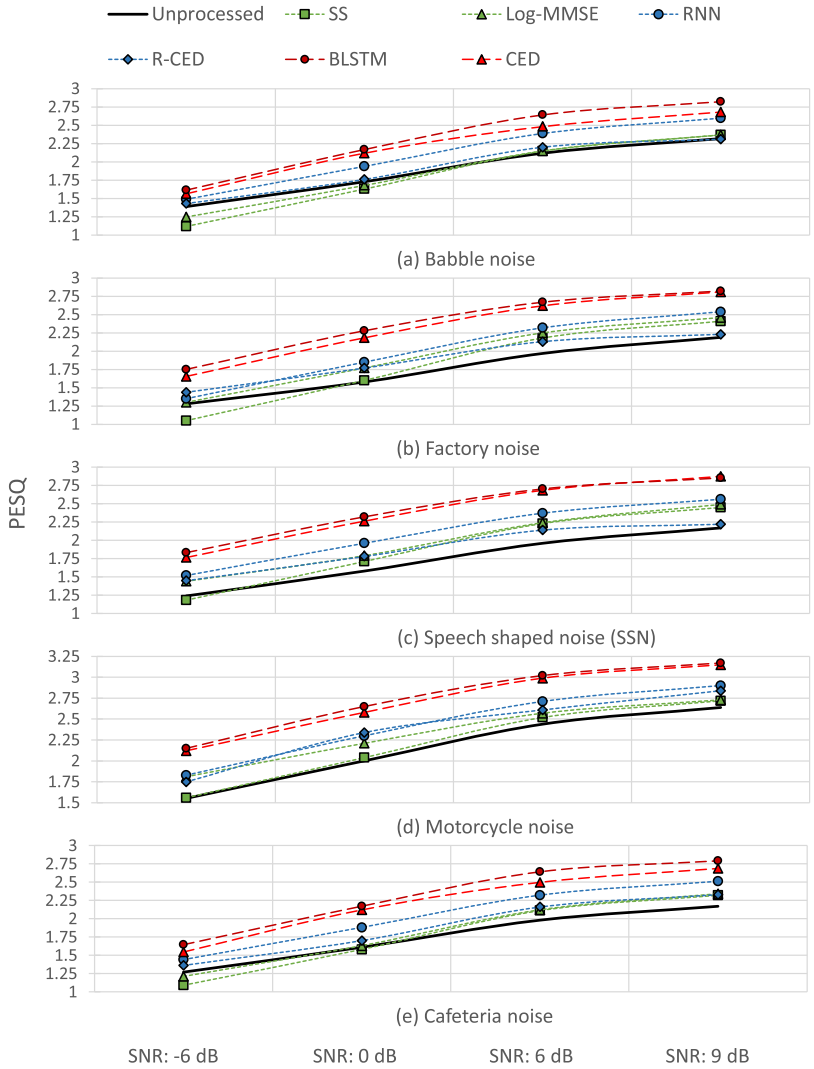

Fig. 4. PESQ comparison between the proposed methods (*BLSTM* and *CED*) and baseline traditional methods (*SS* Boll, 1979 and *Log-MMSE* Ephraim and Malah, 1985) and baseline DNN-based methods (*RNN* and *R-CED* Park and Lee, 2016) for different noise types and SNRs.

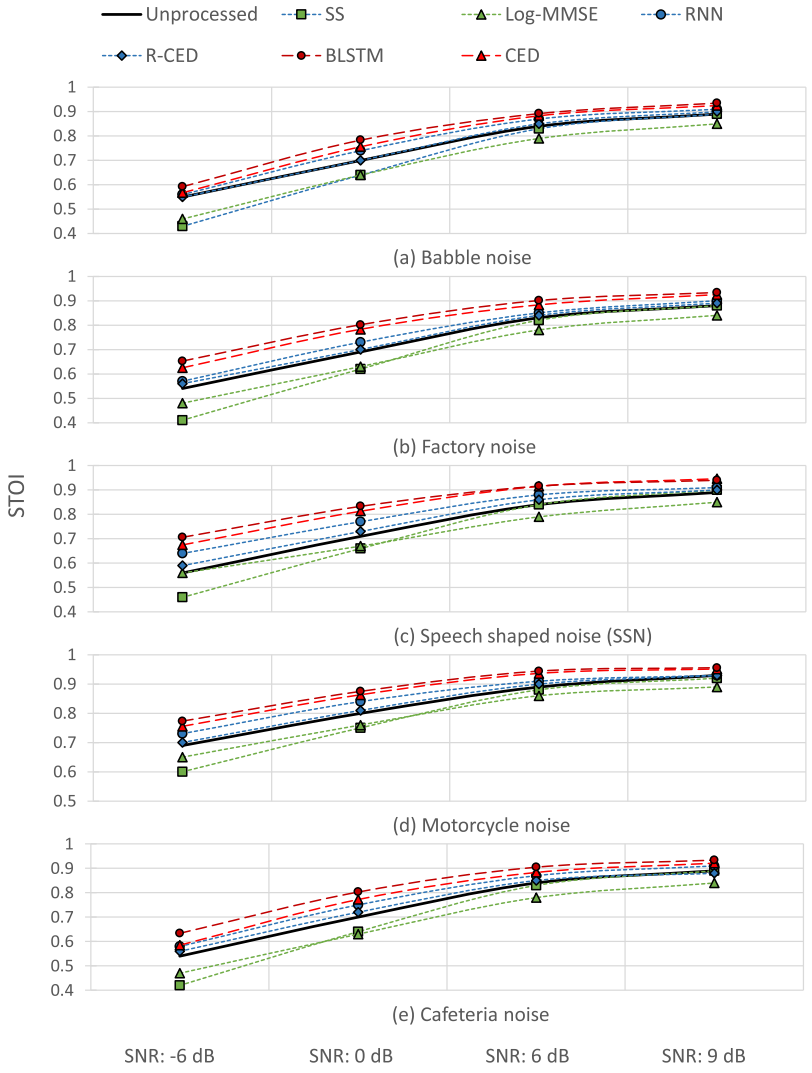

Fig. 5. STOI comparison between the proposed methods (*BLSTM* and *CED*) and baseline traditional methods (*SS* Boll, 1979 and *Log-MMSE* Ephraim and Malah, 1985) and baseline DNN based methods (*RNN* and *R-CED* Park and Lee, 2016) for different noise types and SNRs.

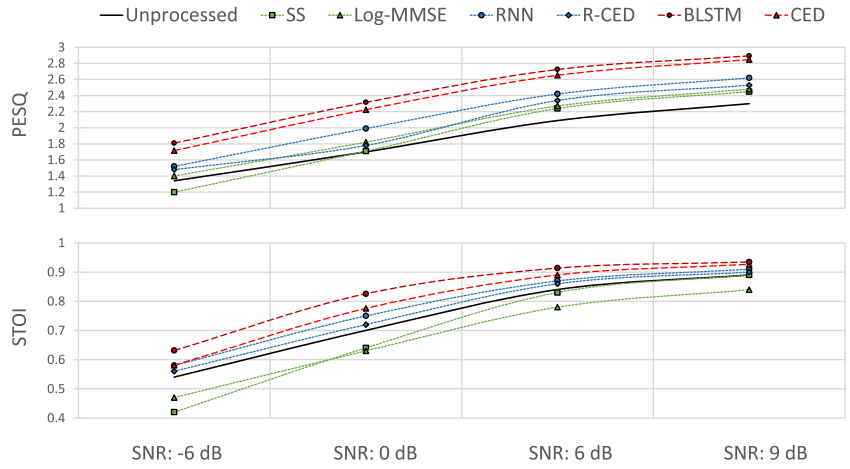

Fig. 6. PESQ and STOI comparisons averaged over all noise types.

Fig. 7. PESQ and STOI comparisons averaged over all noise types for different numbers of hidden units (64, 128, 256, 512 and 1024) per layer in the BLSTM network.

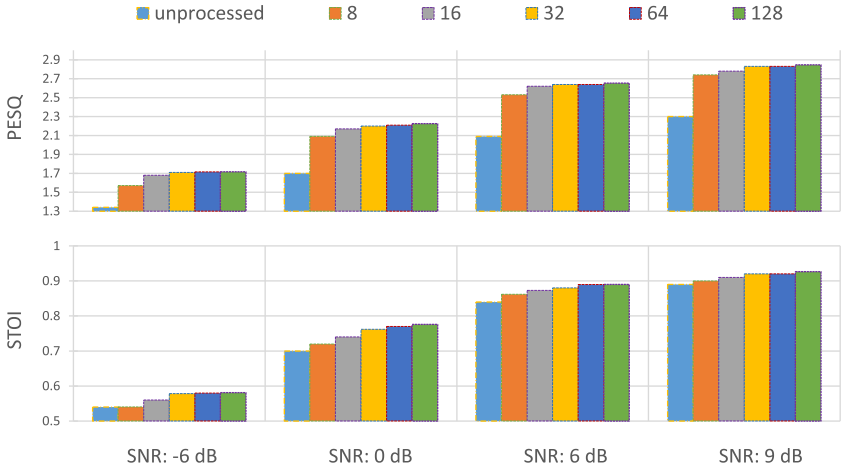

Fig. 8. PESQ and STOI comparisons averaged over all noise types for different numbers of filters (M = 8, 16, 32, 64 and 128) in the first convolutional layer in the CED network. The numbers of filters in the other convolutional and deconvolutional layers are powers-of-two times of M, following the same symmetric pattern shown in Fig. 2.

spectrogram as the input feature to the networks. The main difference between these two inputs is the frequency resolution. Compared to the linear-frequency scale, mel-frequency scale has a better correspondence with human auditory systems. It has a higher frequency resolution at low frequencies but a lower frequency resolution at high frequencies. The PESQ and STOI results are shown in Fig. 12. From the results, we can see that there is a slight difference between the two types of input. The log-amplitude linear frequency spectrogram yields slightly better PESQ and STOI results, therefore, we selected it as our input in other experiments.

4.4. Application in automatic speaker verification

In this section, first we describe the ASV system used for the experiments, and then we use the different speech enhancement methods as a pre-processor for the described ASV system and compare their

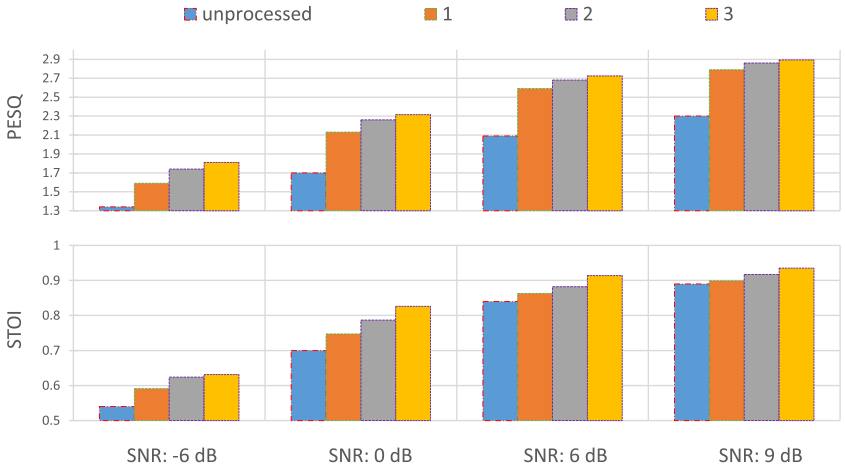

Fig. 9. PESQ and STOI comparisons averaged over all noise types for different numbers of hidden layers (1, 2 and 3) in the BLSTM network.

Fig. 10. PESQ and STOI comparisons averaged over all noise types for different numbers of layers (3, 5 and 7) in the CED network.

Fig. 11. PESQ and STOI comparisons averaged over all noise types for mean-squared error (MSE) and binary cross-entropy (BCE) loss functions in BLSTM and CED networks.

effects in decreasing the verification error rate.

The i-vector approach is the state of the art in speaker verification and is commonly used in current commercial systems. Therefore, we evaluate our SE system on an i-vector-based text-independent ASV system with probabilistic linear discriminant analysis (PLDA) scoring, which is implemented based on Larcher et al. (2016), an open source Python library for speaker and language recognition. We choose this open-source ASV implementation for result reproduction purposes. For all ASV experiments, we use 13 Mel Frequency Cepstrum Coefficients (MFCCs) with their delta and double-delta features, resulting in a 39-dimension vector. The rank of the T matrix, and therefore the dimension of the i-vectors, is set to 100. We found that using low dimensional i-vectors provide better EER results when the utterances are short in duration. We apply length normalization described in (Garcia-Romero and Espy-Wilson, 2011). The dimensionalities of the subspaces F and G in PLDA training are set to 100 × 50 and 100, respectively.

We use the widely used metric, equal error rate (EER), to evaluate the ASV performance. EER is defined as the intersection point where false rejection rate and false acceptance rate are equal. Lower EER means better ASV performance.

4.4.1. Datasets

We run our experiments on two datasets: *VBG RANDNUM* and *RedDots*. All of the utterances in both datasets are sampled at 8 kHz, and are natural noisy utterances with a high SNR.

VBG RANDNUM is a dataset from the Voice Biometrics Group (VBG)'s production system. It contains 1300 English utterances from 100 speakers, where each speaker has 3 enrollment utterances, and 10 verification utterances. Please note that in our experiments we use multi-session scoring described in (Lee et al., 2013). Each utterance contains four random digits and its average length is 6.3 s. We estimated the SNR of *VBG RANDNUM* samples using the tool described in (Vondrasek and Pollk, 2005) with a window size of 8 ms and 50% overlap, and show the SNR distribution in Fig. 13. We use the enrollment and verification samples of 50 speakers to train the ASV system, namely, the UBM, T matrix and PLDA parameters. These samples already contain natural noise, but we also added artificial noise between 10–25 dB SNR level to 100 randomly chosen samples to obtain a multi-condition training set. We use the remaining 50 speakers for evaluation, where there are in total 50 (target speakers) × 10 (verification utterances) × 50 (potential speakers) = 25,000 trials in the evaluation. Since this dataset contains constrained speech, we follow the general guidelines described in (Bimbot et al., 2004) and keep the number of components used for the UBM small (128 components). Some examples from the *VBG RANDNUM* corpus and their enhanced versions are available for the research community.[2]

This *VBG RANDNUM* dataset is representative of VBG's RandomPIN™offering, which is currently deployed (commercially) in 8

[2] Free download at http://www.ece.rochester.edu/projects/wcng/code.html.

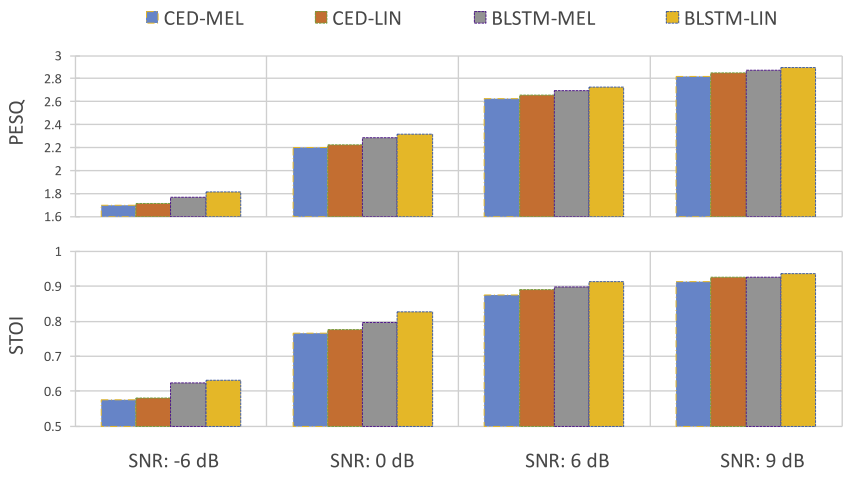

Fig. 12. PESQ and STOI comparisons averaged over all noise types between log-mel spectrogram (MEL) and log-linear spectrogram (LIN) inputs for BLSTM and CED networks.

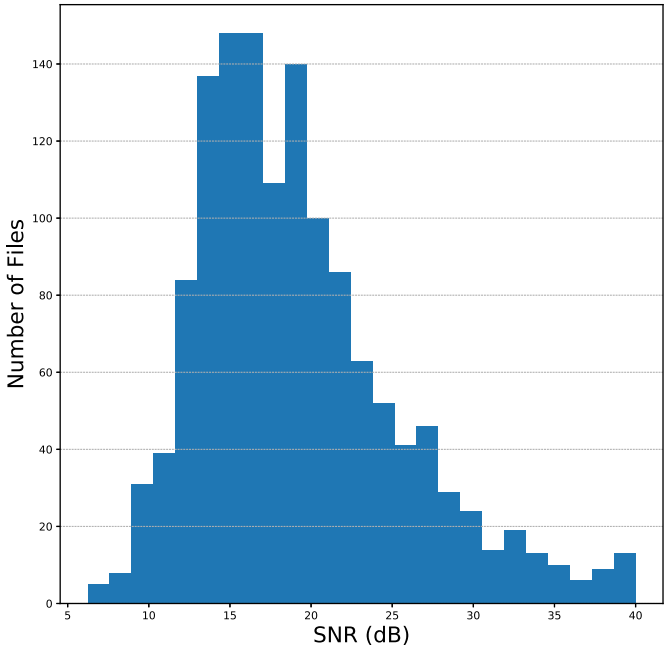

Fig. 13. Histogram of SNR estimation of *VBG RANDNUM* files.

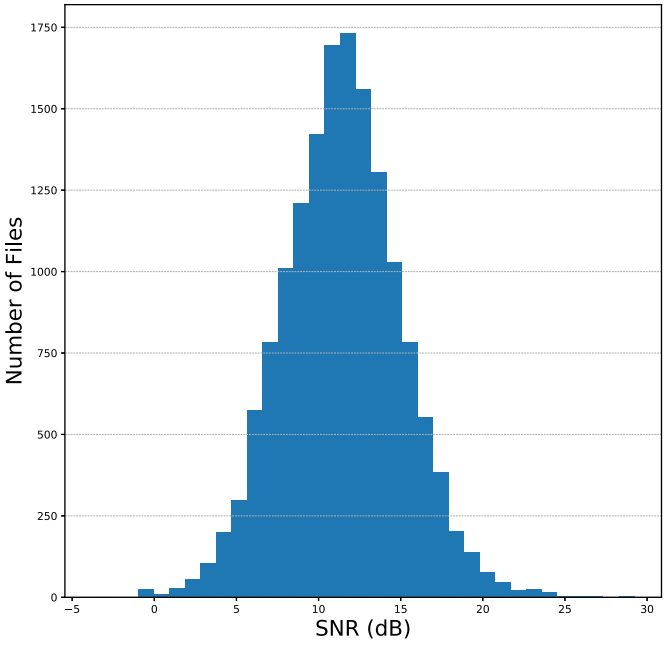

Fig. 14. Histogram of SNR estimation of *RedDots* files.

countries, using 36 different languages. VBG is currently processing over 6 million RandomPIN™verification requests annually, and this number is growing rapidly.

To build voice-prints for RandomPIN™, users are prompted to repeat a series of six separate static numeric digit phrases, each five digits long. Note that each RandomPIN™user is prompted with these same enrollment phrases. To verify the speaker, the VBG system generates a random 4- or 5-digit phrase for the user to repeat.

VBG uses text-independent technology to perform speaker verification. As a pre-processor, VBG uses automatic speech recognition to make sure all content is spoken as requested. A variety of audio quality assessment tests are also performed to ensure the audio is of sufficient quality to perform biometric voice processing. Should samples fail content or quality pre-checks as part of a verification request, the system will automatically generate a new random PIN and re-prompt the user.

Using constrained data (digits only) helps the client to create reliable voice-prints in a limited amount of time efficiently. As the majority of VBG's customers are interactive voice response (IVR) users (standalone or as an entry to a call center conversation), telephone connect

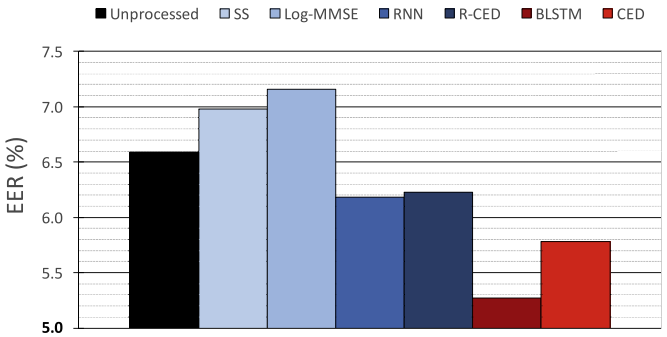

Fig. 15. *VBG RANDNUM* EER results. Note that the y-axis starts from 5.0%.

time (i.e., "call handle time") becomes a sufficient economic consideration. Thus, shorter and more compact uses of voice biometrics are advantageous. Moreover, when RandomPIN™is combined with other security factors, such as knowledge-based authentication (KBA), an extremely reliable match can be provided to VBG clients - without the

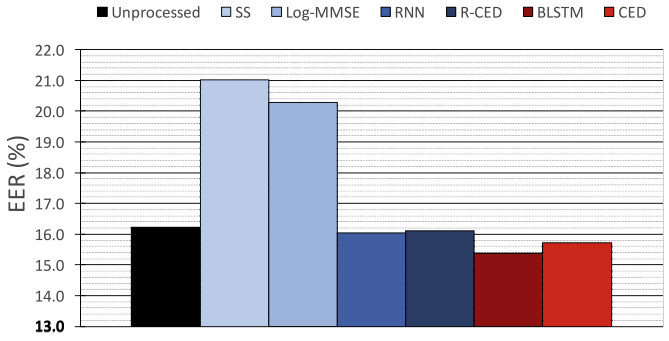

Fig. 16. EER results for *RedDots* Dataset. Note that the y-axis starts from 13.0%.

lengthy data collection requirement of free speech or passive voice biometric applications (which VBG also supports commercially).

The second dataset, namely *RedDots* (Lee et al., 2015), is a collection of short utterances in English from native and non-native speakers reading text prompts to mobile devices. The sessions are collected over a long period (aimed to be over a year), where each speaker records a session per week. The dataset contains 13 female and 49 male speakers from different regions worldwide, a total of 21 countries, which results in vast inter-speaker variations. Since the data collection is carried out from a mobile device, the user can choose to record an utterance in any place, indoor or outdoor. Therefore utterances contain various types of noise with various SNRs. We estimated the SNR of the *RedDots* samples in the same way that we estimated the SNR of the *VBG RANDNUM* samples. Fig. 14 shows the SNR distribution for the RedDots samples.

We conduct our experiments in a text-independent fashion. Therefore, we use *RedDots part 04: text-independent* test set. There are a total of 136,698 target trials and 5,098,950 imposter trials for males, and 26,928 target and 184,368 imposter trials for females in this test set. Since the number of female samples are relatively limited in this dataset, we only use male trials in our experiments, different from the gender-independent case in the experiments with the *VBG RANDNUM* dataset.

To conduct a more comprehensive evaluation in different noise conditions, we also mix *RedDots* test utterances with five types of noise at SNRs of −6, 0, 6 and 9 dB to create more noisy utterances and report their ASV results. To construct the UBM and i-vector models (i.e., the *T* matrix), we use two other datasets, NIST SRE06 (NIS, 2006) and the NIST SRE08 (NIS, 2008). We randomly draw 650 male speakers from these datasets' training set. We also added artificial noise between 10–25 dB SNR level to 150 randomly chosen samples to obtain a multi-condition training set. Since the test samples are unconstrained speech, we set the number of mixtures in the UBM to 2048 in our experiments, as suggested in (Bimbot et al., 2004). Finally, we used the remaining male data in the *RedDots* dataset that is not included in the trials to train PLDA parameters.

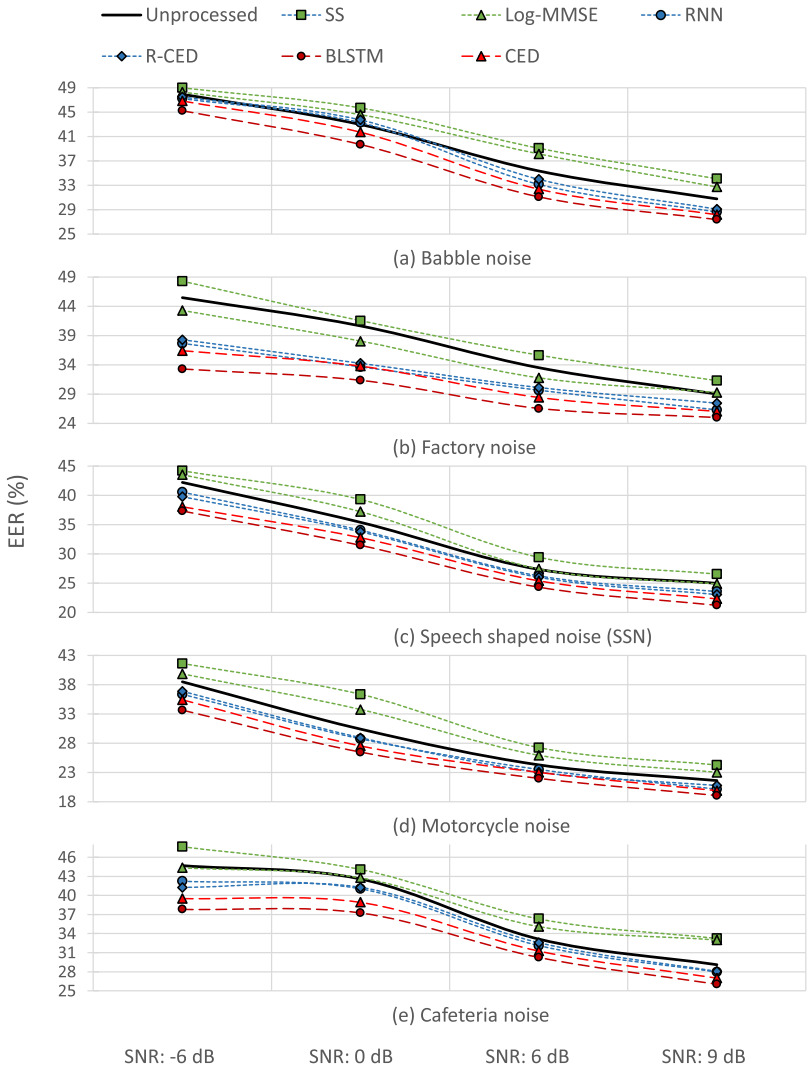

Fig. 17. *RedDots* dataset EER results for different noise types and SNRs.

4.4.2. Evaluations

Figs. 15 and 16 show the EER results for speech that is unprocessed as well as speech that is enhanced with *SS, Log-MMSE, RNN, R-CED, CED* and *BLSTM* for *VBG RANDNUM* and *RedDots* datasets, respectively.

For constrained speech data, Fig. 15 (*VBG RANDNUM*) shows that *BLSTM* significantly decreases the EER compared to other techniques, from the unprocessed EER (%) result of 6.59 to 5.21. This is followed by *CED* with an EER (%) value of 5.78. The gap between *BLSTM* and *CED* EER results are significant, although their PESQ and STOI values shown in Figs. 4–6 are close. While the reason for this mismatch is unclear, this result suggests that speech quality and intelligibility measures for speech enhancement preprocessing modules only provide qualitative predictions of the final speaker verification error rates. *RNN* yields slightly better results compared to *R-CED*, which is consistent with the PESQ and STOI results. An important observation from these results is that there is a benefit of using DNN-based approaches as a front-end SE module since all DNN-based methods yield EER improvements on naturally noisy data. *SS* and *Log-MMSE*, however, significantly increases EER, showing that they cannot deal with non-stationary noise conditions well.

The same trends can be observed for unconstrained speech data (*RedDots*) results shown in Fig. 16, although the improvement on EER of the DNN-based methods are slighter compared to the *VBG RANDNUM* results. *SS* and *Log-MMSE*, again, do not perform well in this dataset.

Fig. 17 shows the artificial test results, i.e., the EER results when additional noise is introduced at an SNR of −6, 0, 6 and 9 dB to the *RedDots* dataset. For all noise cases, the SS method increases the EER. The Log-MMSE method yields EER improvements in low SNRs for factory and cafeteria noise types, however, it does not provide EER improvements for all the other noise types and SNRs. The DNN-based methods yield EER improvements in most cases. The BLSTM network performs the best for all noise types.

5. Conclusions

In this work, two DNN-based speech enhancement methods (BLSTM and CED) are introduced, and their effect as a preprocessor for an automatic speaker verification (ASV) system is investigated. Compared to two classical and two DNN-based speech enhancement baselines, the proposed methods significantly improve the PESQ and STOI of the enhanced speech on different kinds of non-stationary noise that are unseen in the training data. Moreover, they decrease the verification error rate on natural utterances encountered by the verification system and on utterances artificially mixed with additional noise. We show that all DNN-based methods investigated in this work yield performance improvements when they are used as a front-end noise removal module on natural noisy data collected from real customers, while the classical methods degrade the performance in the same conditions.

Acknowledgments

The authors would like to thank the three anonymous reviewers for their thorough and constructive comments that have significantly improved this paper. This research was partially supported by the Voice Biometrics Group (VBG) and the National Science Foundation grant No. 1617107.

References

Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-García, J., Petrovska-Delacrétaz, D., Reynolds, D.A., 2004. A tutorial on text-independent speaker verification. EURASIP J. Appl. Signal Processing. 2004, 430–451.

Boll, S., 1979. Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans. Acoust. 27 (2), 113–120.

Chen, J., Wang, Y., Yoho, S.E., Wang, D., Healy, E.W., 2016. Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises. J. Acoust. Soc. Am. 139 (5), 2604–2612.

Chollet, F., et al., 2015. Keras. https://github.com/fchollet/keras.

Dehak, N., Kenny, P.J., Dehak, R., Dumouchel, P., Ouellet, P., 2011. Front-end factor analysis for speaker verification. IEEE Trans. Audio Speech Lang. Process. 19 (4), 788–798.

Duan, Z., Mysore, G.J., Smaragdis, P., 2012. Speech enhancement by online non-negative spectrogram decomposition in nonstationary noise environments. Thirteenth Annual Conference of the International Speech Communication Association.

Ephraim, Y., Malah, D., 1984. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. IEEE Trans. Acoust. 32 (6), 1109–1121.

Ephraim, Y., Malah, D., 1985. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. IEEE Trans. Acoust. 33 (2), 443–445.

Erdogan, H., Hershey, J.R., Watanabe, S., Le Roux, J., 2015. Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks. Proc. ICASSP. pp. 708–712.

Garcia-Romero, D., Espy-Wilson, C.Y., 2011. Analysis of i-vector length normalization in speaker recognition systems. Interspeech. 2011. pp. 249–252.

Garofolo, J.S., Lamel, L.F., Fisher, W.M., Fiscus, J.G., Pallett, D.S., 1993. Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1. NASA STI/Recon Technical Report n 93.

Godin, K.W., Sadjadi, S.O., Hansen, J.H., 2013. Impact of noise reduction and spectrum estimation on noise robust speaker identification. INTERSPEECH. pp. 3656–3660.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Huang, P.-S., Kim, M., Hasegawa-Johnson, M., Smaragdis, P., 2015. Joint optimization of masks and deep recurrent neural networks for monaural source separation. IEEE/ACM Trans. Audio Speech Lang. Process. 23 (12), 2136–2147.

Jiang, Y., Lee, K.A., Tang, Z., Ma, B., Larcher, A., Li, H., 2012. Plda modeling in i-vector and supervector space for speaker verification. Thirteenth Annual Conference of the International Speech Communication Association.

Kenny, P., Boulianne, G., Dumouchel, P., 2005. Eigenvoice modeling with sparse training data. IEEE Trans. Speech Audio Process. 13 (3), 345–354.

Kjems, U., Boldt, J.B., Pedersen, M.S., Lunner, T., Wang, D., 2009. Role of mask pattern in intelligibility of ideal binary-masked noisy speech. J. Acoust. Soc. Am. 126 (3), 1415–1426.

Kolbœk, M., Tan, Z.-H., Jensen, J., 2016. Speech enhancement using long short-term memory based recurrent neural networks for noise robust speaker verification. Proc. SLT. IEEE, pp. 305–311.

Larcher, A., Lee, K.A., Meignier, S., 2016. An extensible speaker identification sidekit in python. Proc. ICASSP. IEEE, pp. 5095–5099.

Lee, K.A., Larcher, A., Guangsen, W., Patrick, K., Brummer, N., van Leeuwen, D., Aronowitz, H., Kockmann, M., Vaquero, C., Ma, B., Li, H., Stafylakis, T., Alam, J., Swart, A., Perez, J., 2015. The RedDots data collection for speaker recognition. pp. 2996–3000.

Lee, K.-A., Larcher, A., You, C.H., Ma, B., Li, H., 2013. Multi-session PLDA scoring of i-vector for partially open-set speaker detection. Proc. INTERSPEECH. pp. 3651–3655.

Li, Y.P., Wang, D.L., 2009. On the optimality of ideal binary time-frequency masks. Speech Commun. 51 (3), 230–239.

Loizou, P.C., 2013. Speech Enhancement: Theory and Practice. CRC press.

Lu, X., Matsuda, S., Hori, C., Kashioka, H., 2012. Speech restoration based on deep learning autoencoder with layer-wised pretraining. Proc. INTERSPEECH.

Lu, X., Tsao, Y., Matsuda, S., Hori, C., 2013. Speech enhancement based on deep denoising autoencoder. Proc. INTERSPEECH. pp. 436–440.

Lu, X., Tsao, Y., Matsuda, S., Hori, C., 2014. Ensemble modeling of denoising autoencoder for speech spectrum restoration. Proc. INTERSPEECH. 14. pp. 885–889.

Narayanan, A., Wang, D.L., 2013. Ideal ratio mask estimation using deep neural networks for robust speech recognition. Proc. ICASSP. pp. 7092–7096.

Nilsson, M., Soli, S.D., Sullivan, J.A., 1994. Development of the hearing in noise test for the measurement of speech reception thresholds in quiet and in noise. J. Acoust. Soc. Am. 95 (2), 1085–1099.

The nist year 2006 speaker recognition evaluation plan, 2006. https://catalog.ldc.upenn.edu/docs/LDC2011S10/sre-06_evalplan-v9.pdf.

The nist year 2008 speaker recognition evaluation plan, 2008. https://catalog.ldc.upenn.edu/docs/LDC2011S07/sre-08_evalplan-0408.doc.

Panayotov, V., Chen, G., Povey, D., Khudanpur, S., 2015. Librispeech: an asr corpus based on public domain audio books. Proc. ICASSP. pp. 5206–5210.

Park, S.R., Lee, J., 2016. A fully convolutional neural network for speech enhancement. Comput. Res. Repository abs/1609.07132.

Pirker, G., Wohlmayr, M., Petrik, S., Pernkopf, F., 2011. A pitch tracking corpus with evaluation on multipitch tracking scenario. Proc. INTERSPEECH. pp. 1509–1512.

Prince, S.J., Elder, J.H., 2007. Probabilistic linear discriminant analysis for inferences about identity. Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on. IEEE, pp. 1–8.

Reynolds, D.A., Quatieri, T.F., Dunn, R.B., 2000. Speaker verification using adapted gaussian mixture models. Digit. Signal Process. 10 (1–3), 19–41.

Rix, A., Beerends, J., Hollier, M., Hekstra, A., 2001. Perceptual evaluation of speech quality (pesq), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs. ITU-T Recommendation 862.

Sound ideas, 2018. https://www.sound-ideas.com/.

Srinivasan, S., Roman, N., Wang, D.L., 2006. Binary and ratio time-frequency masks for robust speech recognition. Speech Commun. 48 (11), 1486–1501.

Taal, C.H., Hendriks, R.C., Heusdens, R., Jensen, J., 2011. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. IEEE Trans. Audio Speech Lang. Process. 19 (7), 2125–2136.

Varga, A., Steeneken, H.J., 1993. Assessment for automatic speech recognition: ii. noisex-92: a database and an experiment to study the effect of additive noise on speech recognition systems. Speech Commun. 12 (3), 247–251.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., 2010. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J. Mach. Learn. Res. 11 (Dec), 3371–3408.

Vondrasek, M., Pollk, P., 2005. Methods for speech snr estimation: evaluation tool and

analysis of vad dependency. 14.
Wang, Y., Narayanan, A., Wang, D.L., 2014. On training targets for supervised speech separation. IEEE/ACM Trans. Audio Speech Lang. Process. 22 (12), 1849–1858.
Weninger, F., Erdogan, H., Watanabe, S., Vincent, E., Le Roux, J., Hershey, J.R., Schuller, B., 2015. Speech enhancement with lstm recurrent neural networks and its application to noise-robust asr. Proc. LVA ICA. Springer, pp. 91–99.
Williamson, D.S., Wang, Y., Wang, D.L., 2016. Complex ratio masking for joint enhancement of magnitude and phase. Proc. ICASSP. pp. 5220–5224.
Xu, Y., Du, J., Dai, L.-R., Lee, C.-H., 2014. Dynamic noise aware training for speech enhancement based on deep neural networks. Proc. INTERSPEECH. pp. 2670–2674.
Xu, Y., Du, J., Dai, L.-R., Lee, C.-H., 2014. An experimental study on speech enhancement based on deep neural networks. IEEE Signal Process. Lett. 21 (1), 65–68.
Xu, Y., Du, J., Dai, L.-R., Lee, C.-H., 2014. Global variance equalization for improving deep neural network based speech enhancement. Proc ChinaSIP. pp. 71–75.
Xu, Y., Du, J., Dai, L.-R., Lee, C.-H., 2015. A regression approach to speech enhancement based on deep neural networks. IEEE/ACM Trans. Audio Speech Lang. Process. 23 (1), 7–19.
Zeiler, M.D., Krishnan, D., Taylor, G.W., Fergus, R., 2010. Deconvolutional networks. Proc. CVPR. IEEE, pp. 2528–2535.
Zhao, X., Shao, Y., Wang, D., 2011. Robust speaker identification using a casa front-end. Proc. ICASSP. pp. 5468–5471.
Zhao, X., Wang, Y., Wang, D., 2014. Robust speaker identification in noisy and reverberant conditions. IEEE/ACM Trans. Audio Speech Lang. Process. 22 (4), 836–845.