A PHONEME-BASED PRE-TRAINING APPROACH FOR DEEP NEURAL NETWORK WITH APPLICATION TO SPEECH ENHANCEMENT

Shlomo E. Chazan, Sharon Gannot and Jacob Goldberger

Faculty of Engineering, Bar-Ilan University, Ramat-Gan, 5290002, Israel

ABSTRACT

In this study, we present a new phoneme-based deep neural network (DNN) framework for single microphone speech enhancement. While most speech enhancement algorithms overlook the phoneme structure of the speech signal, our proposed framework comprises a set of phoneme-specific DNNs (pDNNs), one for each phoneme, together with an additional phoneme-classification DNN (cDNN). The cDNN is responsible for determining the posterior probability that a specific phoneme was uttered. Concurrently, each of the pDNNs estimates a phoneme-specific speech presence probability (pSPP). The speech presence probability (SPP) is then calculated as a weighted averaging of the phoneme-specific pSPPs, with the weights determined by the posterior phoneme probability. A soft spectral attenuation, based on the SPP, is then applied to enhance the noisy speech signal. We further propose a compound training procedure, where each pDNN is first *pre-trained* using the phoneme labeling and the cDNN is trained to classify phonemes. Since these labels are unavailable in the test phase, the entire network is then trained using the noisy utterance, with the cDNN providing phoneme classification. A series of experiments in different noise types verifies the applicability of the new algorithm to the task of speech enhancement. Moreover, the proposed scheme outperforms other schemes that either do not consider the phoneme structure or use simpler training methodology.

Index Terms— neural network, phoneme, deep learning.

1. INTRODUCTION

Deep learning methods have gained significant popularity in recent years, due to their success in various classification tasks, as compared with classic algorithms. More recently, DNNs were introduced to the field of speech processing, and in particular, speech enhancement.

A DNN was trained as a deep auto-encoder (DAE) in [1, 2] to enhance noisy speech. The DAE was trained with noisy inputs and clean targets in order to learn the nonlinear filter from the noisy inputs to the clean targets. The results might suffer from speech distortion when introduced with an untrained noise.

Different methods were presented by Wang et al. to find the ideal binary mask (IBM) [3, 4] which represents the frequencies in which the speech is active, where '1' represents speech and '0' represents noise. Given an IBM, a spectral subtraction is then applied to mitigate the noise. The results proved sufficient for improving the performance of automatic speech recognition (ASR) systems but suffer from artifacts in speech enhancement tasks, mainly due to the applied hard decision [5]. The ideal ratio mask (IRM) was then proposed as a soft mask for speech enhancement [6]. This method significantly improved the speech quality, yet the performance with an unfamiliar noise types was not always satisfactory.

A hybrid approach, borrowing concepts from both model-based and the DNN-based approaches, was presented in [7]. In this paper, the phoneme structure of the speech signal was utilized to improve the speech enhancement. A Mixture of Gaussians (MoG) representation of the phonemes was constructed, and a DNN was used as a phoneme classifier. The SPP was accurately estimated using both a generative and discriminative methodologies.

The phoneme information of the speech was also utilized in [8] with a full discriminative approach. Forty DNNs were trained, one for each phoneme, to find the IRM. In the test phase, given a noisy utterance, an ASR system is used to detect the correct phoneme label. Only the DNN associated with the identified phoneme is activated to predict the IRM and to apply the enhancement. Yet, if the ASR system produces wrong labels, the enhancement will deteriorate, since wrong DNN may be used. This may result in significant speech quality degradation, considering ASR performance in noisy environment.

In this paper, we present a joint system which combines the phoneme-specific DNNs (pDNNs) with a phoneme classifier (also implemented by a DNN). The noisy input is simultaneously processed by the phoneme-classification DNN and the pDNNs, to jointly provide SPP estimates. We also describe a joint training scheme using *pre-training* approach for initialization.

2. PROBLEM FORMULATION AND ENHANCEMENT STRATEGY

Let $z(t)$ be the observed noisy speech, where $x(t)$ and $y(t)$ denote the speech and noise signals, respectively.

$$z(t) = x(t) + y(t). \tag{1}$$

Let $Z(n,k)$ denote the short-time Fourier transform (STFT) of $z(t)$, with n the frame index and $k = 0, \ldots, L-1$ the frequency index. The frame length is set to L, and the overlap between successive frames is set to $3L/4$ samples. Denote the log-spectral vector at frame n by $\mathbf{z}(n)$ with its k-th frequency component defined by:

$$Z_k(n) = \log|Z(n,k)|,\ k = 0, \ldots, L/2. \tag{2}$$

Similarly, define the respective log-spectral vectors of the clean speech and the noise signal by $\mathbf{x}(n)$ and $\mathbf{y}(n)$.

Following Nádas et al. [9], the noisy log-spectral vector can be approximated by:

$$\mathbf{z}(n) \approx \max(\mathbf{x}(n), \mathbf{y}(n)) \tag{3}$$

such that the maximization is component-wise over the elements of $\mathbf{x}(n)$ and $\mathbf{y}(n)$. The maximization approximation was used for speech recognition [9], speech enhancement [10, 11, 7] and speech separation tasks [12, 13].

978-1-5090-2007-2/16/$31.00

Define the SPP $\rho_k(n) \in [0,1]$ as the probability given the noisy utterance $\mathbf{z}(n)$ that the k-th frequency component of the n-th noisy frame is dominated by speech. Given the SPP, the k-th bin of the log-spectrum of the clean speech $\hat{X}_k(n)$, is estimated using soft attenuation:

$$\hat{X}_k(n) = Z_k(n) - (1 - \rho_k(n)) \cdot \beta \tag{4}$$

where β is the noise attenuation level (in the log domain). In our implementation we set β to correspond to attenuation of 20 dB, which yielded high noise suppression while maintaining low speech distortion. Respectively, in vector form:

$$\hat{\mathbf{x}}(n) = \mathbf{z}(n) - (\mathbf{1} - \boldsymbol{\rho}(n)) \cdot \beta \tag{5}$$

where $\mathbf{1}$ is a vector of ones with the same dimensions as $\boldsymbol{\rho}(n)$, the vector concatenation of $\rho_k(n),\ k = 0, 1, \ldots, L/2$.

The observed noisy phase is used for reconstructing the time-domain speech signal similarly to most speech enhancement algorithms. In this paper we propose a framework based on a set of DNNs to estimate the SPP with a novel training scheme, consisting a phoneme-based pre-training and a global training stage for the entire network.

3. ENHANCEMENT ALGORITHM

3.1. SPP-based Enhancement Procedure: Overview

Each phoneme is characterized by a distinct structure in the log-spectral domain. Thus, it makes sense to tailor a separate enhancement procedure for each phoneme. Naturally, the phoneme label is unknown during the enhancement procedure, and should be estimated. In our approach, the phoneme classification task is implemented by a DNN. Concurrently, a set of deep neural networks (DNNs) is utilized to estimate the phoneme-specific speech presence probability (pSPP) vector. Finally, the pSPP decisions are averaged using the phoneme classification results to obtain the (global) SPP vector that is used for the actual enhancement as described in (4).

We first describe the phoneme classifier utilizing a DNN responsible for the estimating the phoneme probabilities given the log-spectral vector of the noisy speech $\mathbf{z}(n)$:

$$p_i(n) = p(I(n) = i | \mathbf{z}(n); \text{cDNN}),\ i = 1, \ldots, m \tag{6}$$

where $I(n)$ is a random variable depicting the phoneme label and m is the number of phonemes. Concurrently, $\mathbf{z}(n)$ is fed to a set of pSPP estimators:

$$\rho_{i,k}(n) = p(X_k(n) > Y_k(n) | \mathbf{z}(n), I(n) = i; \text{pDNN}_i) \tag{7}$$

with pDNN_i the DNN associated with the i-th phoneme. The global SPP decision for the k-th frequency band $\rho_k(n)$ is obtained by merging all the phoneme-based decisions by the following weighted averaging:

$$\rho_k(n) = p(X_k(n) > Y_k(n) | \mathbf{z}(n)) = \sum_{i=1}^{m} p_i(n) \cdot \rho_{i,k}(n). \tag{8}$$

We use the same architecture for all the pDNNs, each of which has two hidden layers with 1000 rectified linear unit (ReLU) neurons. The output layer provides a soft SPP decision for each frequency band, implemented by a sigmoid function.

Fig. 1 depicts a block diagram of the proposed compound system with the cDNN, m pDNNs, and a soft attenuation block.

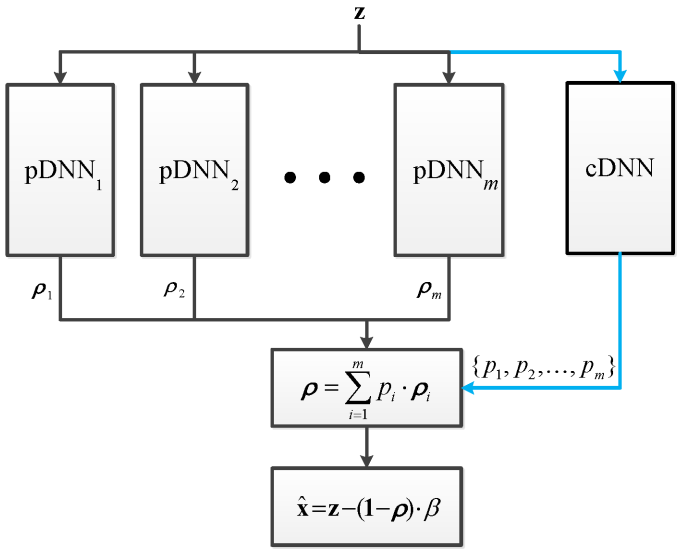


Fig. 1: Block diagram of the multiple DNNs algorithm.

3.2. Training

For the proposed scheme we need to train both the cDNN and the m pDNNs. Phoneme classification is a standard task in speech recognition that can be trained independently of our enhancement goals.

3.2.1. Phoneme-classification DNN

The first step in the training stage is to train the cDNN. We applied a standard deep neural network [7], consisting of two hidden layers constructed by ReLU neurones and output transfer function set to the *softmax* function.

The TIMIT corpus [14], which is a phoneme-labeled database, was used for training. The input vector was set to be the 39 mel-frequency cepstral coefficients (MFCC) features of the current frame with additional 16 context frames. Overall, the input vector dimension is $39 \times 17 = 663$. In the training phase, clean speech utterances are used as inputs to the DNN, while in the test phase, the inputs to the phoneme classifier are noisy speech utterances. To alleviate this mismatch, we used a standard preprocessing stage for robust phoneme classification, namely cepstral mean and variance normalization (CMVN) [15]. During the training phase, we used the dropout method to avoid over-fitting [16]. Once the classifier is trained, its weights are kept fixed. After convergence, the cDNN has reached 74.6% success rate in the phoneme classification task, tested on the clean test set of the TIMIT corpus.

3.2.2. Phoneme-specific DNN

To train the pDNN the true SPP values are required. To construct such a dataset, clean speech signals were contaminated by a (single type) noise signal with a pre-defined signal to noise ratio (SNR) level. The log-spectrum of the synthesized noisy signals were used as the input to the network. Let $W_k(n)$ be the k-th log-spectrum of the simulated noisy data at frame n and $X_k(n)$ and $Y_k(n)$ are the respective clean speech and noise signal. The corresponding SPP

binary targets $c_k(n) \in \{0,1\}$ are set according to:

$$c_k(n) = \mathbb{I}_{\{X_k(n) > Y_k(n)\}} \tag{9}$$

where $\mathbb{I}$ is the indicator operator. Further define the target vector at frame n, $\mathbf{c}(n) = [c_0(n), \ldots, c_{L/2}(n)]^\top$, and the SPP vector at frame n, $\boldsymbol{\rho}(n) = [\rho_0(n), \ldots, \rho_{L/2}(n)]^\top$. The loss-function is defined as the sum of the square errors of the SPP decisions over all time indexes:

$$L = \sum_n \|\mathbf{c}(n) - \boldsymbol{\rho}(n)\|^2 = \sum_n \|\mathbf{c}(n) - \sum_{i=1}^{m} p_i(n)\boldsymbol{\rho}_i(n)\|^2 \tag{10}$$

where $\boldsymbol{\rho}_i(n)$ is the SPP decisions based on pDNN$_i$. The model parameters are $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m\}$, where $\boldsymbol{\theta}_i$ is the parameter set of DNN$_i$.

If the phoneme labels $I(n) \in \{1, \ldots, m\}$ are known, then each pDNN can be separately trained by the following back-propagation equations:

$$\frac{\partial L}{\partial \boldsymbol{\theta}_i} = 2\sum_n \mathbb{I}_{\{I(n)=i\}} (\boldsymbol{c}(n) - \boldsymbol{\rho}(n))^\top \cdot \frac{\partial \boldsymbol{\rho}_i(n)}{\partial \boldsymbol{\theta}_i}. \tag{11}$$

In our case, we only have a phoneme label estimation based on the noisy speech. Hence, the back-propagation algorithm divides the classification error among the phonemes as follows:

$$\frac{\partial L}{\partial \boldsymbol{\theta}_i} = 2\sum_n \underbrace{p(I(n)=i|\mathbf{z}(n))}_{p_i(n)} \cdot (\boldsymbol{c}(n) - \boldsymbol{\rho}(n))^\top \cdot \frac{\partial \boldsymbol{\rho}_i(n)}{\partial \boldsymbol{\theta}_i}. \tag{12}$$

We can deduce from (12) that the contribution of a noisy frame $\mathbf{z}(n)$ to the training of pDNN$_i$ is proportional to the probability that the phoneme i was uttered at time n.

3.3. Parameter Initialization

DNN weights are often randomly initialized. In our case we have encountered convergence problems with this random initialization. This may be attributed to the significant amount of phoneme classification errors, impeding the pDNNs from proper training.

We therefore propose to utilize the phoneme labels, provided by the TIMIT database [14], in order to initialize the training. We first train each pDNN independently (using (11)), as proposed in [8]. To train the i-th pDNN, we construct a sub-database by artificially adding a single noise type to speech frames associated with phoneme i.

This training strategy is based on the assumption that phoneme labels are known. At the test phase, we cannot assume the availability of such information. To circumvent this mismatch between the train and test phases, we have decided to use the above procedure only as a *pre-training* stage of the pDNNs rather than the conventional random initialization. Note, that standard pre-training is an unsupervised method that ignores the labels in the training data and is based on a greedy layer-wise procedure using either restricted Boltzmann machines (RBMs) [17] or noisy auto-encoders [18]. In our case the pre-training is based on additional information, namely the phoneme labels, that is only available in the training phase but not in test phase. This additional information is used to initialize the parameters by separately training each pDNN.

After the pre-training, we continue with the entire set of DNNs and jointly train them by minimizing the loss-function using (12). Hence, all classification errors of the phoneme classifier will be included in the training phase. We stress that the phoneme classifier is not trained again and we continue to use the network weights as described in 3.2.1.

4. EXPERIMENTAL STUDY

4.1. Experiment setup

To test the proposed algorithm we have contaminated speech signal with several types of noise from NOISEX-92 database [19], namely *Speech-like*, *Babble*, and *Factory*. The noise was added to the clean signal drawn from the test set of the TIMIT database (24-speaker core test set), with 5 levels of SNR at -5 dB, 0 dB, 5 dB, 10 dB and 15 dB in order to represent various real-life scenarios.

In order to evaluate the performance of the proposed algorithm we have also implemented two other variants of the proposed algorithm. The first comprises only one DNN trained to estimate the SPP from noisy inputs without separate treatment for phonemes. This implementation is denoted *no-phonemes*. In the second variant, denoted *hard decision*, each pDNN was separately trained, skipping the joint training stage. Here, the SPP $\boldsymbol{\rho}(n)$ was set to the pSPP of the most probable phoneme:

$$\boldsymbol{\rho}(n) = \boldsymbol{\rho}_l(n) \tag{13}$$

where $l = \text{argmax}_i\, p_i(n) = p(I(n) = i|\mathbf{z}(n)),\ i = 1, 2, \ldots, m$.

To assess the performance of the proposed algorithm we have used the objective *composite measure*, proposed by Hu and Loizou [20]. The composite measure weights the log likelihood ratio (LLR), the perceptual evaluation of speech quality (PESQ) [21] and the weighted spectral slope (WSS) [22] to predict the *rating* of the background distortion (Cbak), the speech distortion (Csig) and the overall quality (Covl) performance. The rating is based on the 1-5 mean opinion score (MOS) scale, with clean speech signal achieving MOS value of 4.5.

4.2. The training parameters

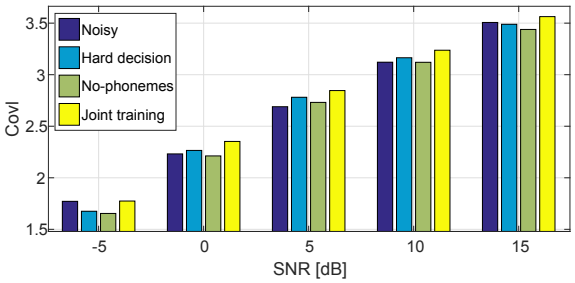
The training method was already discussed in Sec 3.2. The number of phonemes was set to $m = 39$. We used 8 context frames, 4 from the past and 4 from the future, for the training of the pDNNs. Overall the input vector size is $9 * 257 = 2313$, and the target vector size is 257. The noise type used in the training phase was speech-like noise drawn from NOISEX-92 database [19] with input SNR of 15 dB.

4.3. Phoneme classification results

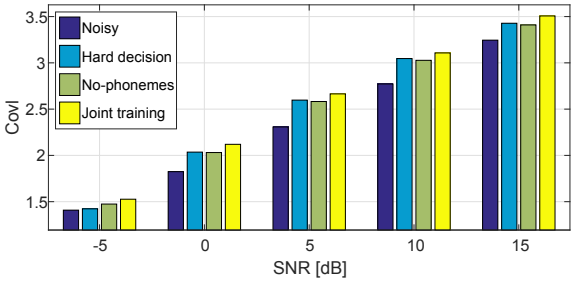
The cDNN is a crucial part of our algorithm. Table 1 introduces the percentage of correct classification results obtained for the noisy test data. The percentage of correct classification for the clean signal is 74.6%. It is evident that at low SNRs the percentage of correct classification is quite low. Therefore, we expect that the hard decision approach will activate the incorrect pDNN, leading to poor enhancement quality. Consequently, it is crucial that the training system will take these classification errors into account.

Table 1: Phoneme classification performance in various noise types.

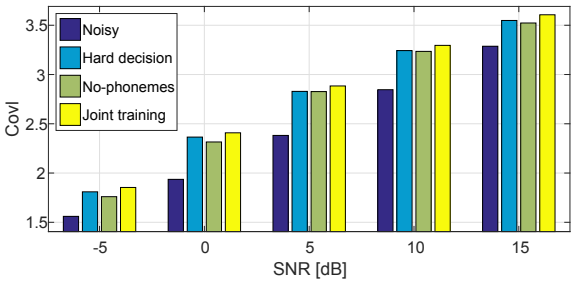
Noise type	**-5 dB**	**0 dB**	**5 dB**	**10 dB**	**15 dB**
Factory	29%	37%	47%	56%	64%
Babble	34%	44%	52%	59%	65%



(a) Babble noise.



(b) Factory noise.



(c) Speech noise.

Fig. 2: Covl results for various noise types and SNR levels.

4.4. Objective results

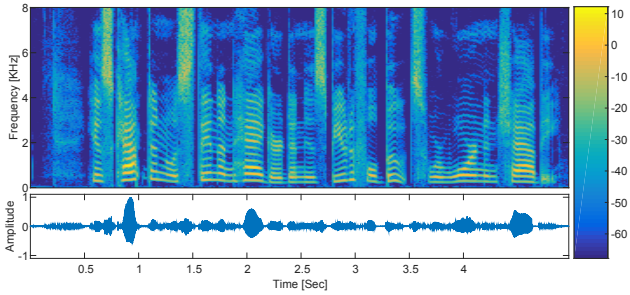
Fig. 2 depicts the Covl results for all examined algorithms for the Babble, Factory and Speech-like noise types as a function of the input SNR. It is evident that the proposed algorithm outperforms the other variants. Interestingly, this also applies to the speech-like noise used for the training of all variants.

4.5. Sonograms

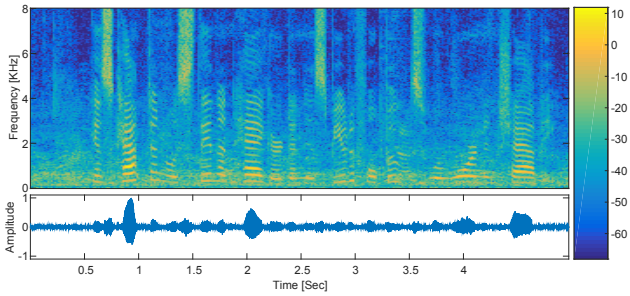
In this paper we aim at finding the SPP, $\boldsymbol{\rho} = p(\mathbf{x} > \mathbf{y}|\mathbf{z})$. In order to illustrate the performance of our SPP estimator, a clean signal was contaminated by Factory noise with SNR=10dB. Note that the system did not train on this noise type. Fig. 3 depicts the clean and noisy signals together with the estimated SPP $\boldsymbol{\rho}$ and the enhanced signal. It can be observed that $\boldsymbol{\rho}$ adequately track the speech signal.

5. CONCLUSIONS

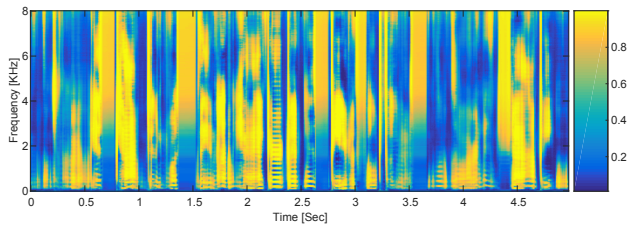
In this paper we presented a novel speech enhancement algorithm. The system utilizes the phoneme structure of the speech by training a different DNN for each phoneme. A trained phoneme classifier is used to estimate the phoneme probabilities. The phoneme probabilities and the phoneme-specific speech presence probability are combined to estimate the SPP. A compound training scheme with pre-training of the pDNN and a joint training of the entire set of DNNs is also proposed. A series of experiments in various noise types and levels have shown the advantages of the proposed scheme in comparison to simplified variants.



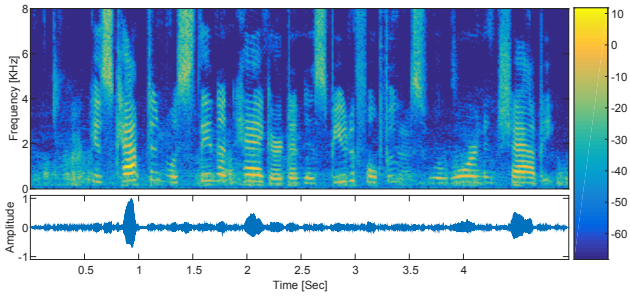
(a) Clean.



(b) Noisy.



(c) $\boldsymbol{\rho} = p(\mathbf{x} > \mathbf{y}|\mathbf{z})$.



(d) Enhanced signal.

Fig. 3: Sonograms of the enhancement with Factory noise SNR=10dB.

6. REFERENCES

[1] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, Jan. 2014.

[2] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder." in *INTERSPEECH*, 2013, pp. 436–440.

[3] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[4] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[5] D. Wang, U. Kjems, M. S. Pedersen, J. B. Boldt, and T. Lunner, "Speech intelligibility in background noise with ideal binary time-frequency masking," *The Journal of the Acoustical Society of America*, vol. 125, no. 4, pp. 2336–2347, 2009.

[6] S. Srinivasan, N. Roman, and D. Wang, "Binary and ratio time-frequency masks for robust speech recognition," *Speech Communication*, vol. 48, no. 11, pp. 1486–1501, 2006.

[7] S. E. Chazan, J. Goldberger, and S. Gannot, "A hybrid approach for speech enhancement using mog model and neural network phoneme classifier," *CoRR*, vol. abs/1510.07315, 2015. [Online]. Available: http://arxiv.org/abs/1510.07315

[8] Z.-Q. Wang, Y. Zhao, and D. Wang, "Phoneme-specific speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 146–150.

[9] A. Nádas, D. Nahamoo, and M. Picheny, "Speech recognition using noise-adaptive prototypes," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 37, no. 10, pp. 1495–1503, Oct. 1989.

[10] D. Burshtein and S. Gannot, "Speech enhancement using a mixture-maximum model," *IEEE Trans. on Speech and Audio Processing*, vol. 10, no. 6, pp. 341–351, Sep. 2002.

[11] Y. Yeminy, S. Gannot, and Y. Keller, "Speech enhancement using a multidimensional mixture-maximum model," in *International Workshop on Acoustic Echo and Noise Control (IWAENC)*, 2010.

[12] S. T. Roweis, "One microphone source separation," in *Neural Information Processing Systems (NIPS)*, vol. 13, 2000, pp. 793–799.

[13] M. Radfar and R. Dansereau, "Single-channel speech separation using soft mask filtering," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2299–2310, Nov. 2007.

[14] J. S. Garofalo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "The DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM," Linguistic Data Consortium, Tech. Rep., 1993.

[15] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 745–777, Apr. 2014.

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

[17] G. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, 2006.

[18] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2007.

[19] A. Varga and H. J. Steeneken, "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 communication*, vol. 12, no. 3, pp. 247–251, 1993.

[20] Y. Hu and P. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, Jan. 2008.

[21] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, 2001, pp. 749–752.

[22] D. Klatt, "Prediction of perceived phonetic distance from critical-band spectra: A first step," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 7, May 1982, pp. 1278–1281.