# FEATURE DENOISING FOR SPEECH SEPARATION IN UNKNOWN NOISY ENVIRONMENTS

*Yuxuan Wang*[1] *and DeLiang Wang*[1,2]

[1]Department of Computer Science and Engineering, The Ohio State University, USA
[2]Center for Cognitive Science, The Ohio State University, USA
{wangyuxu,dwang}@cse.ohio-state.edu

## ABSTRACT

Speech separation has been recently formulated as a classification problem. Classification as a form of supervised learning usually performs well on background noises when parts of them are seen in the training set. However, the performance can be significantly worse when generalizing to completely unseen noises. In this study, we present a method that alleviates the generalization issue by attempting to denoise acoustic features before training and testing. We show that a standard multilayer perceptron with proper regularization performs well on this task. Experimental results indicate that the resulting separation system performs significantly better in a variety of unknown noises in low SNR conditions. In a negative SNR condition, we also show that the proposed system produces more intelligible speech according to two recently proposed objective speech intelligibility measures.

***Index Terms***— Speech separation, feature denoising, generalization, deep neural networks

## 1. INTRODUCTION

Single-channel speech separation is a persistent challenge for decades. A series of recent studies (e.g., [1, 2, 3]) has shown that speech separation can be effectively formulated as a binary classification problem, in which the computational goal is to estimate the ideal binary mask (IBM). The IBM is ideal in the sense that it is constructed from premixed target and interference, where 1 indicates that the target energy exceeds the interference energy by a local signal-to-noise (SNR) criterion (LC) in the corresponding time-frequency (T-F) unit, and 0 otherwise. Specifically,

$$IBM(t,f) = \begin{cases} 1, \text{ if } SNR(t,f) > LC \\ 0, \text{ otherwise,} \end{cases}$$

where $SNR(t,f)$ denotes the local SNR (in decibels) within the T-F unit at time $t$ and frequency $f$. For separation, one extracts acoustic features within each T-F unit and train proper classifiers to decide the target-dominance of the T-F unit.

Classification-based speech separation is a form of supervised learning, and thus speech separation can greatly benefit from the modeling power of machine learning techniques. However, supervised learning models run the general risk of not generalizing well to mismatched test conditions. This is also true for classification-based speech separation, which performs worse in unknown noise conditions unless the classifiers are trained on large datasets. How to improve generalization with limited training data, and how to make the system perform better even with large amounts of training data are the questions that we tackle in this paper.

The generalization issue is mainly caused by mismatched feature patterns or distributions between training and test sets. A fundamental cause is that we lack environment-invariant speech features. To deal with this issue, we propose a data-driven approach that learns a (nonlinear) mapping of noisy features to a more stable and coherent feature space. Specifically, we design a neural network in which the input is noisy features and the output is the corresponding clean features. This neural network is used to transform raw noisy training data into a new set of training data, which is then combined with raw features for classifier training. In testing, the same network is used to transform the test data. This way the feature distribution of the test data is expected to more likely match that of the training data. Importantly, we have observed that such a feature denoising task is less demanding in terms of generalization, which consequently boosts the performance for the subsequent classification task.

Our work is inspired by the denoising autoencoder (DA) work [4]. DA tries to extract more invariant features by adding a small amount of noise (e.g., salt-and-pepper noise) to raw data. The model is trained to reconstruct the raw data using the noisy data, and the hidden layer activations are used as learned features. Our model differs from DA in that (1) we deal with real noises of interest rather than artificial noises; (2) our model functions as a feature denoiser rather than an autoencoder as we are interested in network outputs rather than hidden activations; and (3) the input and output need not to be the same kind of feature and may have different dimensions. Methods exist that map noisy features

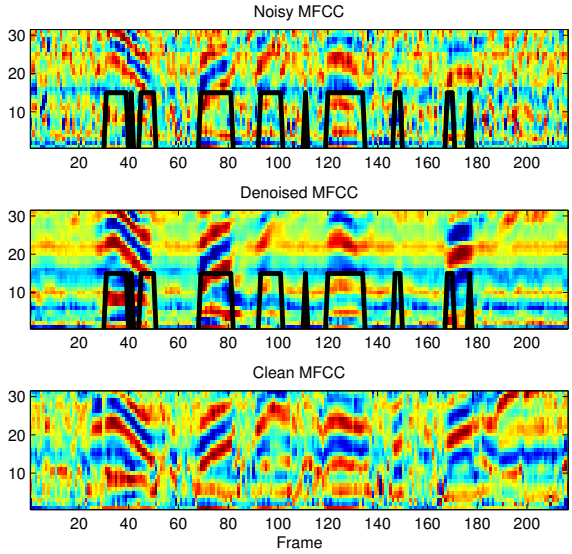


**Fig. 1**. An example of feature denoising for an IEEE utterance mixed with an unseen factory noise at 0 dB. Top panel: noisy MFCC features in a filter channel. Middle panel: corresponding denoised MFCC features. Bottom panel: corresponding clean MFCC features. The black solid line indicates which T-F units are target-dominant (i.e. with a label of 1).

to more robust ones. Representative methods include mean variance normalization plus ARMA filtering (MVA) [5], histogram equalization [6], and the SPLICE algorithm [7]. The differences between previous works and our work lie in the underlying task, the denoising method, and the type of feature to be denoised (i.e., unit-level features v.s. frame-level features).

This paper is organized as follows. We briefly introduce the framework of classification-based speech separation in the next section. Section 3 discusses the proposed feature denoising method, followed by experimental results in Section 4. Section 5 concludes this paper.

## 2. CLASSIFICATION-BASED SPEECH SEPARATION

The computational goal of classification-based speech separation is to estimate the IBM, which can substantially improve speech intelligibility for both normal-hearing and hearing-impaired listeners (see e.g. [8]). We use a 64 channel gammatone filterbank as our analysis frontend. Specifically, the noisy mixtures are passed to the gammatone filterbank with center frequencies ranging from 50 to 8000 Hz. We form a T-F representation called cochleagram [9] by windowing (we use a 20-ms window with a 10-ms overlapping) the output from each filter channel. We extract acoustic features and train subband classifiers for individual filter channels, with the IBM providing training labels.

Since a binary decision needs to be made for each T-F unit, we extract acoustic features from the subband signal slice underlying each T-F unit. These unit-level features are more suitable than traditional frame-level ones for the separation task [10]. In [10], we also proposed to use a combination of amplitude modulation spectrogram (AMS), relative spectral transform and perceptual linear prediction (RASTA-PLP) , and mel-frequency cepstral coefficients (MFCC) as the feature vector for each T-F unit. As for subband classifier, in our recent work we found that pretrained deep neural networks (DNN) tend to outperform previously used Gaussian mixture models (GMM) and support vector machines (SVM) [11].

## 3. FEATURE DENOISING

### 3.1. Motivation

As described above, we aim to denoise unit-level noisy features into corresponding clean features. We start our discussion with a motivating example shown in Fig. 1. The top panel of Fig. 1 shows 31-D MFCC features in a filter channel, extracted from an IEEE utterance [12] mixed with an unseen factory noise at 0 dB. We can see that the raw MFCC features are very noisy, and some discriminative information is buried in the noise. Nevertheless, if we superimpose the IBM (of this channel) on top of the noisy features, we can see that the target-dominant T-F units tend to have more prominent acoustic patterns. This is illustrated in Fig. 1, where the black solid line indicates target-dominant T-F units. This observation motivates us to design an algorithm that takes a noisy feature as input and attempts to reconstruct its stable structure. The output from this algorithm is expected to be more coherent than the raw feature, thereby improving the generalization performance for subsequent classification.

To this end, we design a multilayer perceptron (MLP), where the input is the noisy feature and the output is the clean feature. As we can see from the middle panel of Fig. 1, the denoised features are closer to the clean ones, and are much more coherent than the noisy ones. Interestingly, for features that are completely dominated by noise, the network tends not to output meaningful structure. This is consistent with an intuitive assumption that unstructured data are difficult to map to structured data, unless there is overfitting. Finally, we point out that the feature denoiser seems to generalize well. The denoiser used in this example is only trained on a speech-shaped noise, and the factory noise is unseen to the denoiser. The denoising network is used to preprocess all the features before training and testing in classification-based separation.

### 3.2. Model design

Two design choices can impact the feature denoising performance. First, we found that using a window of noisy features

as input to denoise the center noisy feature is much more effective than using the single noisy feature alone. Such effectiveness comes from the temporal correlations in speech signal. Second, we found that it is important to properly regularize the denoising network. We achieve the best performance when using a sparsity-inducing regularization on the hidden layer. Specifically, we regularize the mean hidden activations to be a small positive number, such that most of the hidden activations are close to 0 (see e.g. [13]). In this study, we use Kullback-Leibler (KL) divergence to achieve this goal:

$$\mathrm{KL}(\hat{\rho}||\rho) = \sum_k \rho \log \frac{\rho}{\hat{\rho}} + (1-\rho)\log\frac{1-\rho}{1-\hat{\rho}}, \qquad (1)$$

where $k$ indexes hidden units, $\rho$ is the expected mean activation level which we set to 0.01, and $\hat{\rho}$ is the actual averaged hidden activation level of the network. The objective function of the denoising network (for one sample) then becomes:

$$\mathcal{L}(\mathbf{y}, \mathbf{x}; \mathbf{W}) = \frac{1}{2}\|\mathbf{y} - f(\mathbf{x}; \mathbf{W})\|_2^2 + \beta \mathrm{KL}(\hat{\rho}||\rho), \qquad (2)$$

where $\mathbf{x}$ and $\mathbf{y}$ denote the noisy and corresponding clean feature, respectively. $f(\cdot)$ is the nonlinear mapping defined by the denoising network, whose parameters are collectively denoted by $\mathbf{W}$. When using multiple hidden layers, we only regularize the last hidden layer. Note that when the hidden layer is set to be overcomplete (i.e., more hidden than input units), the network can be viewed as a nonlinear version of sparse coding. Therefore, it is reasonable to project that the effectiveness of using sparse regularization is due to the observation that clean speech is amenable to overcomplete sparse decomposition [14].

## 4. EXPERIMENTAL RESULTS

Before describing more extensive experiments, we perform a preliminary evaluation to verify the utility of feature denoising. The denoiser is trained on 100 IEEE utterances mixed with a speech-shaped noise at 0 dB. The denoiser uses only one hidden layer. As mentioned in Sec. 2, we use two hidden layer deep neural networks pretrained by restricted Boltzmann machines (RBM) as the subband classifiers for classification-based separation. These DNNs are trained on 50 IEEE utterances mixed with 12 nonspeech noises at 0 dB. The test mixtures are created by mixing 20 new utterances with 10 unseen noises, also at 0 dB. We choose unit-level MFCC as the raw feature, which does not generalize well to unseen noises [10]. We consistently found that the best performance is achieved when concatenating the denoised feature with the raw feature. Therefore we only present denoising results using this concatenated feature.

We document results using classification accuracy, overall HIT−FA, voiced-interval HIT−FA, and unvoiced-interval HIT−FA. The HIT−FA criterion is suggested by Kim et al. [1], and shown to be well correlated to human speech intelligibility. The HIT rate is the percent of correctly classified target-dominant T-F units (1's) in the IBM, and the FA rate is the percent of wrongly classified interference-dominant T-F units (0's). Table 1 shows the performance comparisons between using raw MFCC features (RAW) and raw features concatenated with denoised features (RAW+DNS). It is clear that by making use of denoised features, we can achieve significant improvements over 10 unseen noises. The performance can be further boosted by using delta features derived from the denoised features (RAW+DNS+Δ). As shown in Table 1, overall we achieve about 3.3 percentage improvements in accuracy and 12 percentage improvements in HIT−FA, which is quite encouraging given that the denoiser is only trained on one noise. We have also experimented with deep denoising network that is pretrained by RBMs. However, we only achieve about 0.5% HIT−FA improvement while the

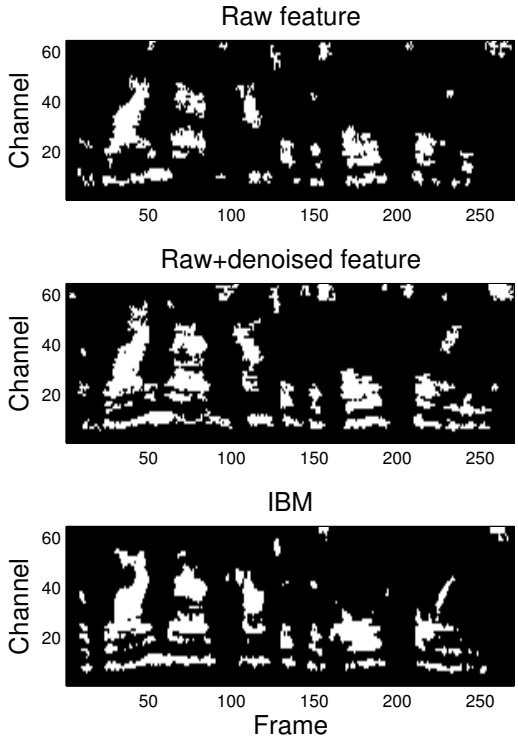


**Fig. 2**. Masks obtained on an IEEE utterance mixed with an unseen speech-shaped noise at 0 dB.

**Table 1**. Comparisons by using MFCC as raw feature

| Feature | Accuracy | Overall HIT−FA | Voiced HIT−FA | Unvoiced HIT−FA |
|---|---|---|---|---|
| RAW | 82.8% | 54.9% | 57.8% | 38.2% |
| RAW+DNS | 85.1% | 61.3% | 64.1% | 46.4% |
| RAW+DNS+Δ | 86.1% | 66.9% | 67.1% | 61.4% |

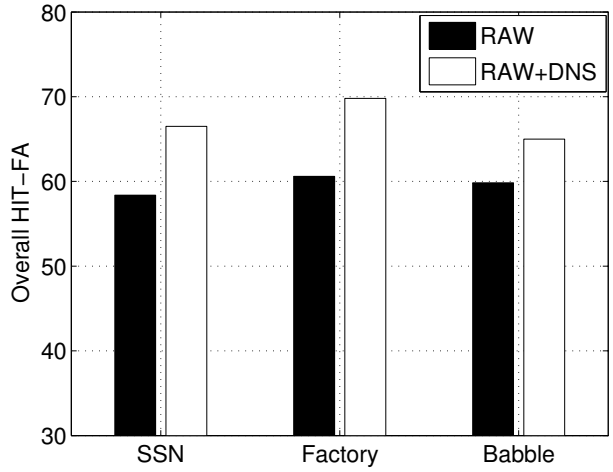


**Fig. 3**. Overall HIT−FA comparisons when using a complementary feature set as raw features.

training becomes significantly slower. Hence, we use one hidden layer denoising network (with random initializations) in the rest of the experiments.

Next, we show that the denoising network also works for more robust features. We choose the complementary feature set, AMS+RASTA-PLP+MFCC, as the raw features [10], and test 20 utterances mixed with three unseen challenging noises (speech-shaped, factory, and babble noise) at 0 dB. The denoiser and subband DNNs are trained on 25 noises, excluding the three test noises. As can be seen in Fig. 3, we still achieve about 7 percentage HIT−FA improvements averaged over 3 noises. Fig. 2 shows estimated masks on a test utterance mixed with the unseen speech-shaped noise.

We now present results in the -5 dB input SNR condition. The test mixtures are obtained by mixing 10 IEEE utterances with 12 unseen broadband noises[1] at -5 dB. Due to the difficulty of this task, both denoiser and subband DNNs are trained on 100 nonspeech noises [15], and the complementary feature set is used. Note that we set $LC = -10$ dB. We point out that in the -5 dB SNR condition, human speech intelligibility is no longer perfect (see e.g., [1]). Therefore, aside from HIT−FA, we also use a recently proposed short-time objective intelligibility measure (STOI) [16] as an evaluation metric. STOI has been shown to be highly correlated with human speech intelligibility scores.

Table 2 shows the comparisons in terms of overall HIT−FA and STOI. The average STOI of the mixtures and IBM separated speech is 0.61 and 0.81, respectively. Since DNNs can also produce a posterior probability of target-dominance for each T-F unit, we evaluate STOI using both estimated binary mask and posterior (soft) mask. As can be seen, using denoised features significantly outperforms raw features for both HIT−FA and STOI. The best STOI result is obtained by the estimated posterior masks using RAW+DNS+$\Delta$ features.

[1]We use a speech-shaped noise and 11 noises from the NOISEX corpus: white, pink, HF channel, babble, factory 1, factory 2, jet 1, jet 2, destroyer engine, destroyer operations, and F-16 noise.

**Table 2**. Objective intelligibility measure comparisons on -5 dB mixtures. Average STOI of mixtures: 0.61

| System/Feature | Overall HIT−FA | STOI (binary) | STOI (posterior) |
|---|---|---|---|
| Speech Enhancement [17] | n/a | n/a | 0.59 |
| RAW | 52% | 0.64 | 0.65 |
| RAW+DNS+$\Delta$ | 60% | 0.67 | 0.70 |
| IBM | n/a | 0.81 | n/a |

We also compare with a recently proposed speech enhancement algorithm [17]. As shown in the table, this algorithm produces lower STOI scores than the unprocessed mixtures.

## 5. CONCLUDING REMARKS

We have proposed a denoising neural network where the input is unit-level noisy features and output is the corresponding clean features. The denoising network is used to preprocess features before training and testing in classification-based speech separation systems. We have shown that using denoised features significantly boosts performance in unknown noisy conditions, in terms of classification accuracy and the two objective intelligibility measures of HIT−FA and STOI.

One may consider using the denoising network to directly denoise frame-level features and convert the denoised spectra/cepstra to time domain signals. However, we have not obtained good results using this approach, perhaps because frame-level features do not generalize as well as unit-level ones [10].

We will consider using other types of neural network to denoise frame-level features. The results presented in this study are still preliminary. For example, although we have shown that the RBM pretrained deep denoiser does not significantly outperform the randomly initialized shallow denoiser, we do not rule out this possibility if other pretraining methods are used. For example, denoising autoencoder seems to be a more natural choice than RBM.

## 6. ACKNOWLEDGEMENTS

This research was supported in part by an AFOSR grant (FA9550-12-1-0130), an NIDCD grant (R01 DC012048), an STTR subcontract from Kuzer, and the Ohio Supercomputer Center.

## 7. REFERENCES


[1] G. Kim, Y. Lu, Y. Hu, and P.C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *Journal of the Acoustical Society of America*, vol. 126, pp. 1486–1494, 2009.

[2] Z. Jin and D. Wang, "A supervised learning approach to monaural segregation of reverberant speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, pp. 625–638, 2009.

[3] D. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Speech Separation by Humans and Machines*, Divenyi P., Ed., pp. 181–197. Kluwer Academic, Norwell MA., 2005.

[4] P. Vincent, H. Larochelle, Y. Bengio, and P.A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. ICML*, 2008, pp. 1096–1103.

[5] C.P. Chen and J.A. Bilmes, "MVA processing of speech features," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, pp. 257–270, 2007.

[6] A. De La Torre, A.M. Peinado, J.C. Segura, J.L. Pérez-Córdoba, M.C. Benítez, and A.J. Rubio, "Histogram equalization of speech representation for robust speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 13, pp. 355–366, 2005.

[7] L. Deng, A. Acero, L. Jiang, J. Droppo, and X. Huang, "High-performance robust speech recognition using stereo training data," in *Proc. ICASSP*, 2001, pp. 301–304.

[8] D.S. Brungart, P.S. Chang, B.D. Simpson, and D. Wang, "Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation," *Journal of the Acoustical Society of America*, vol. 120, pp. 4007–4018, 2006.

[9] D. Wang and G. Brown, Eds., *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*, Hoboken, NJ: Wiley-IEEE Press, 2006.

[10] Y. Wang, K. Han, and D. Wang, "Exploring monaural features for classification-based speech segregation," *IEEE Trans. Audio, Speech, Lang. Process.*, pp. 270–279, 2013.

[11] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, in press, 2013.

[12] IEEE, "IEEE recommended practice for speech quality measurements," *IEEE Trans. Audio Electroacoust.*, vol. 17, pp. 225–246, 1969.

[13] H. Lee, C. Ekanadham, and A. Ng, "Sparse deep belief net model for visual area V2," in *NIPS*, 2008.

[14] M.S. Lewicki and T.J. Sejnowski, "Learning overcomplete representations," *Neural computation*, vol. 12, pp. 337–365, 2000.

[15] Guoning Hu, "100 nonspeech environmental sounds (http://www.cse.ohio-state.edu/pnl/corpus/HuCorpus.html)," 2004.

[16] C.H. Taal, R.C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of timefrequency weighted noisy speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, pp. 2125–2136, 2011.

[17] R.C. Hendriks, R. Heusdens, and J. Jensen, "MMSE based noise PSD tracking with low complexity," in *Proc. ICASSP*, 2010, pp. 4266–4269.