LEARNING INVARIANT FEATURES FOR SPEECH SEPARATION

Kun Han and *DeLiang Wang*[*]

Department of Computer Science and Engineering
& Center for Cognitive Science
The Ohio State University
Columbus, OH 43210-1277, USA
{hank,dwang}@cse.ohio-state.edu

ABSTRACT

Recent studies on speech separation show that the ideal binary mask (IBM) substantially improves speech intelligibility in noise. Supervised learning can be used to effectively estimate the IBM. However, supervised learning has trouble dealing with the situations where the probabilistic properties of the training data and the test data do not match, resulting in a challenging issue of generalization whereby the system trained under particular noise conditions may not generalize to new noise conditions. We propose to use a novel metric learning method to learn invariant speech features in the kernel space. As the learned features encode speech-related information that is robust to different noise types, the system is expected to generalize to unseen noise conditions. Evaluations show the advantage of the proposed approach over other speech separation systems.

Index Terms— Speech Separation, Domain Adaptation, Kernel Learning, SVM

1. INTRODUCTION

Monaural speech separation is a fundamental problem in speech processing where one can only utilize intrinsic properties of a sound mixture to separate the target speech from the masker. Researchers have attempted to solve monaural speech separation for decades. Speech enhancement approaches have been extensively studied, which utilize statistical properties to enhance speech that has been corrupted by non-speech additive noise. Model based approaches rely on pre-trained models to capture the characteristics of individual sound sources for separation. Recent studies in computational auditory scene analysis (CASA) are inspired by human perceptual principles. These efforts so far have achieved limited success.

The ideal binary mask (IBM) has been proposed as a main computational goal for speech separation, which is defined as a binary matrix along time and frequency where a matrix element is 1 if the signal-to-noise ratio (SNR) within the corresponding time-frequency (T-F) unit is greater than a local SNR criterion (LC) and 0 otherwise [1]. Previous studies have demonstrated that the IBM leads to large improvement on speech intelligibility in noise [2, 3]. To estimate the IBM, one can formulate the problem as binary classification, i.e., a trained classifier decides each T-F unit to be either speech-dominant or interference-dominant based on extracted features.

One issue in supervised classification is that the training data and the test data are expected to extract from the same distribution. When the distribution changes, the trained models may not produce reasonable results in the test dataset. To generalize a speech separation system to unseen noise conditions, one can build a massive training set including a large variety of noises. However, such training is very computationally expensive and it would be impossible to include all noises in a training set.

In this study, we propose to learn invariant speech features in the kernel space using Information-theoretic Metric Learning (ITML) [4]. Because the learned kernel encodes invariant information related only to speech, a classifier trained on this kernel should be able to generalize to unseen noise types. We train support vector machines (SVM) based on the learned kernels and successfully classify test data under new noise conditions. Note that we only consider speech separation from non-speech interference in this study.

In the next section, we relate our approach to existing work on speech separation and metric learning. The overall framework of the system is given in Section 3. Section 4 describes how to learn the kernel and incorporate it into the SVM. We evaluate the system in Section 5 and conclude in Section 6.

2. RELATED WORK

Supervised learning has been recently used to classify T-F units, including multilayer perceptrons (MLP) [5], Gaussian

[*]We thank Brian Kulis for helpful discussion on kernel learning. This research is supported by an AFOSR grant (FA9550-12-1-0130) and an STTR subcontract from Kuzer.

mixture models (GMM) [6], and SVM [7]. These approaches mostly deal with the situations in which the test noises are included in the training set. However, if noises are not seen in the training phase, the probabilistic properties of the extracted features in the test set may differ significantly from those in the training set and the trained models may not work well under these noise conditions.

In machine learning, transfer learning and domain adaptation aim to compensate for data shift, i.e., a change in the feature distribution from the training set to the test set [8]. Relevant methods have been developed in the natural language processing (NLP) [9] and computer vision communities [10, 11, 12], which can be roughly categorized as classifier adaptation and feature transformation. The former approach utilizes the target domain information to adapt the parameters of classifiers [12]. In the speech separation field, Ozerove *et al.* [13] and our previous study [14] utilize noise only intervals to collect noise information for model adaptation. Because the adaptation needs to detect the noise intervals in the test mixtures, it is difficult to apply to real-time processing.

On the other hand, feature transformation utilizes metric learning methods to transfer the input features between domains and then apply a classifier [9, 10, 11]. The advantage of this approach is that the learned features can be domain-independent, which enables it to deal with novel problems with new feature types or dimensionalities [15, 11]. For speech separation, one important property is that the features extracted from speech are usually much more stable than those from noises. In other words, if we can capture the common speech characteristics independent of noise types, it is possible to utilize them to separate speech under various noise conditions. In this paper, we learn invariant speech features across different noise conditions, which allow for generalization to new noises without any prior knowledge of the noise.

3. SPEECH SEPARATION USING KERNEL SVM

3.1. Feature Extraction

An input signal $s(t)$ is first passed through a 64-channel gammatone filterbank spanning from 80 Hz to 5000 Hz. The response of each filter channel is then divided into 20-ms time frames with 10-ms frame shift, forming a cochleagram [16]. We use $u_{c,m}$ to denote a T-F unit for frequency channel c and time frame m. For each T-F unit, we extract acoustic features including amplitude modulation spectrogram (AMS), relative spectral transform and perceptual linear prediction (RASTA-PLP), mel-frequency cepstral coefficients (MFCC), and pitch-based features. Further, for every dimension of the features, we calculate delta features across time frames and frequency channels to capture variation information. The concatenation of these features have been proven to be effective in speech separation [17] and are used in this paper.

3.2. SVM Classification with Learned Kernels

Because of the different spectral properties of speech, we train an SVM in each channel to estimate the IBM. Previous studies directly use extracted features to train the SVM and yield accurate classification results under matched noise conditions [7, 17]. In order to generalize the system to unseen noise conditions, we aim to learn a non-linear transformation $\phi : \mathbb{R}^d \rightarrow \mathbb{R}^{d'}$ to map original features into a high dimensional space, where d and d' denote the dimensionality of the original space and the kernel space respectively. Here, the underlying idea of the feature transformation is that for two data points from different noise conditions (domains), the learned transformation should maximizes the distances between them if they have different labels and minimizes the distances if they have the same label. This class-based cross-domain constraint will be applied during the transformation learning.

Furthermore, because the SVM can be viewed as a kernel machine, instead of explicitly computing $\phi(\mathbf{x})$, we only need to compute a kernel function κ such that $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ [18]. Therefore, we first learn a kernel using data from multiple noise conditions and then apply the learned kernel to the SVM for supervised learning. In the test phase, each data point is also kernelized for classification. We will discuss kernel learning in detail in the next section.

Finally, the SVM labels T-F units in each channel to form an estimated IBM. The separated speech is resynthesized using the cochleagram of the mixture and the estimated IBM [16].

4. DOMAIN-INVARIANT KERNEL LEARNING

4.1. Cross-domain Constraints

In this section, we discuss how to learn domain-invariant features in the kernel space. For a general metric learning problem, given a data set $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n], \mathbf{x}_i \in \mathbb{R}^d$, one aims to learn an appropriate Mahalanobis distance parameterized by a positive definite matrix W between $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$d_W(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T W(\mathbf{x}_i - \mathbf{x}_j) \quad (1)$$

Since W is symmetric positive definite, by factorizing W as $W = G^T G$, we can equivalently view the distance $d_W = ||G\mathbf{x}_i - G\mathbf{x}_j||^2$, that is, the transformation G serves as a linear transformation applying to data points.

Since the linear transformation is not powerful enough for our application, we are interested in working in the kernel space, where we use a non-linear function ϕ to map input into a high-dimensional space. Then, the distance is:

$$d_W(\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)) = (\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j))^T W(\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)) \quad (2)$$

To learn the desired metric, we use the data to create pairwise similarity and dissimilarity constraints. To improve the generalizability in our study, we generate the constraints across different domains based on the labels. Suppose that the training set consists of multiple domains $\mathcal{D}_m, m = 1, \ldots, M$, corresponding to multiple noise conditions, and a data point in the domain $\mathcal{D}_m$ is denoted as $\mathbf{x}_i^{\mathcal{D}_m}$ with its label $y_i^{\mathcal{D}_m}$. To learn the domain-invariant transformation, we use the following cross-domain constraints. For a pair of data points $\mathbf{x}_i$ and $\mathbf{x}_j$ from two different domains $\mathcal{D}_a$ and $\mathcal{D}_b$, we create the constraints:

$$\begin{aligned} &d_W(\phi(\mathbf{x}_i^{\mathcal{D}_a}), \phi(\mathbf{x}_j^{\mathcal{D}_b})) \leq u, \text{ if } y_i^{\mathcal{D}_a} = y_j^{\mathcal{D}_b} \\ &d_W(\phi(\mathbf{x}_i^{\mathcal{D}_a}), \phi(\mathbf{x}_j^{\mathcal{D}_b})) \geq l, \text{ if } y_i^{\mathcal{D}_a} \neq y_j^{\mathcal{D}_b} \end{aligned} \tag{3}$$

where u and l are parameters representing the distance thresholds. As we create cross-domain constraints for every pair of domains, there are totally $\binom{M}{2}$ pairs of domains for constraints.

These cross-domain constraints enforce the algorithm to learn a metric such that the data points with the same label should be close to each other no matter which domains they belong to. By applying the constraints to every pair of domains, the learned transformation captures not only the domain shift between any two of them but also the common information shared by all these domains. Since the data in different domains correspond to speech mixed with different noises, the transformation presumably encodes speech-related information that is independent to noise types.

4.2. Kernel Learning with ITML

Given the constraints in Eq. (3), our problem is to learn a positive-definite matrix W that parameterizes the Mahalanobis distance. We adopt the ITML [4] algorithm and discuss its kernelized version in this subsection. The algorithm uses the LogDet divergence D_{ld} to regularize W against a specified positive definite matrices W_0:

$$D_{ld}(W, W_0) = \text{ trace}(WW_0^{-1}) - \log\det(WW_0^{-1}) \tag{4}$$

and the metric learning problem is:

$$\begin{aligned} &\min_{W \succeq 0} D_{ld}(W, W_0) \\ &\text{s.t. } d_W(\phi(\mathbf{x}_i^{\mathcal{D}_a}), \phi(\mathbf{x}_j^{\mathcal{D}_b})) \leq u, \text{ if } y_i^{\mathcal{D}_a} = y_j^{\mathcal{D}_b} \\ &\quad d_W(\phi(\mathbf{x}_i^{\mathcal{D}_a}), \phi(\mathbf{x}_j^{\mathcal{D}_b})) \geq l, \text{ if } y_i^{\mathcal{D}_a} \neq y_j^{\mathcal{D}_b} \\ &\quad a, b \in \{1, \ldots, M\} \end{aligned} \tag{5}$$

Therefore, we are interested in finding a metric W that is close to an original metric W_0 but satisfies our desired constraints. Note that, we create the constraints for every pair of domains, which is different from previous cross-domain metric learning [10, 11], where only one pair of domains is considered.

We now consider kernelizing the problem. Given a set of data points, let K_0 denote the input kernel matrix for the data, that is, $K_0(i,j) = \kappa_0(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$. In this study, we choose the Gaussian kernel to introduce nonlinearity, i.e., $\kappa_0(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{2\sigma^2})$. We use $K(i,j)$ to denote the kernel we want to learn, i.e., $K(i,j) = \kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T W \phi(\mathbf{x}_j)$. Therefore, according to Eq. (2), we have:

$$\begin{aligned} &d_W(\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)) \\ &= \phi(\mathbf{x}_i)^T W \phi(\mathbf{x}_i) - 2\phi(\mathbf{x}_i)^T W \phi(\mathbf{x}_j) + \phi(\mathbf{x}_j)^T W \phi(\mathbf{x}_j) \\ &= K(i,i) + K(j,j) - 2K(i,j) \end{aligned} \tag{6}$$

In addition, to avoid an infeasible solution in Eq. (5), we incorporate a slack variable $\boldsymbol{\xi}$ to provide a tradeoff between minimizing the divergence between K and K_0 and satisfying the constraints. Finally, the non-linear metric learning problem can be formulated to a kernel learning problem:

$$\begin{aligned} &\min_{K \succeq 0, \boldsymbol{\xi}} D_{ld}(K, K_0) + \gamma D_{ld}(\text{diag}(\boldsymbol{\xi}), \text{diag}(\boldsymbol{\xi}_0)) \\ &\text{s.t. } K(i,i) + K(j,j) - 2K(i,j) \leq \xi_{i,j}, \text{ if } y_i = y_j \\ &\quad K(i,i) + K(j,j) - 2K(i,j) \geq \xi_{i,j}, \text{ if } y_i \neq y_j \\ &\quad (\mathbf{x}_i, y_i) \in \mathcal{D}_a, (\mathbf{x}_j, y_j) \in \mathcal{D}_b, \text{ and } a, b \in \{1, \ldots, M\} \end{aligned} \tag{7}$$

where, γ is the tuning parameter. The entries in $\boldsymbol{\xi}_0$ are set to u for similarity constraints and l for dissimilarity constraints.

To solve this optimization problem, we follow the approach given in [4] which employs Bregman projections to iteratively compute the kernel [11]:

$$K_{t+1} \leftarrow K_t + \beta K_t(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^T K_t \tag{8}$$

where $\mathbf{e}_i$ is the standard basis vector with a 1 in the ith coordinate and β is a parameter computed in the algorithm.

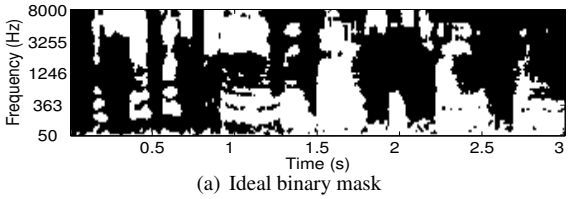
Once we learn the kernel K, it is straightforward to use Eq. (6) to compute the distance between two points $\mathbf{x}_i$ and $\mathbf{x}_j$ that are in the training set. But for new data points $\mathbf{z}_1$ and $\mathbf{z}_2$ that are not in the training set, we need to compute the kernel function $\kappa(\mathbf{z}_1, \mathbf{z}_2)$. Here, we directly give the equation to compute the kernel for a pair of arbitrary data points $\mathbf{z}_1$ and $\mathbf{z}_2$:

$$\kappa(\mathbf{z}_1, \mathbf{z}_2) = \kappa_0(\mathbf{z}_1, \mathbf{z}_2) + \mathbf{k}_1^T K_0^{-1}(K - K_0)K_0^{-1}\mathbf{k}_2 \tag{9}$$

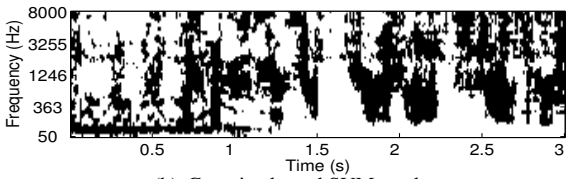
Here, $\mathbf{k}_i = [\kappa_0(\mathbf{z}_i, \mathbf{x}_1), \ldots, \kappa_0(\mathbf{z}_i, \mathbf{x}_n)]^T$, and $\mathbf{x}_i$ is the data point in the training set used to learn the kernel. For details of the kernel learning algorithm, see [4] and [19].

5. EXPERIMENTS

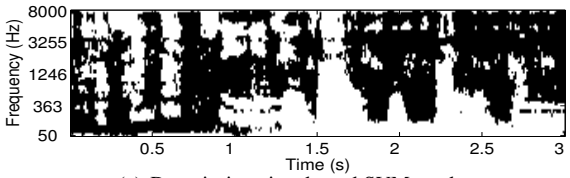
We now evaluate our kernel learning based separation system. The IEEE corpus [20] is used to train and test the system. The input SNR is -5 dB and LC is set to -10 dB, which pose a very challenge problem. To learn the domain-invariant kernel, we first choose 10 utterances mixed with 5 types of noise out of a 100 non-speech noise corpus [21]. Thus, there are around 3,000 data points for each noise condition. We randomly choose a subset of around 100 data points in each condition to create the cross-domain constraints, so $100 \times 100 \times \binom{5}{2} =$



(a) Ideal binary mask



(b) Gaussian kernel SVM mask



(c) Domain-invariant kernel SVM mask

Fig. 1: IBM estimation results. (a) IBM for the mixture. (b) Estimated IBM using the Gaussian kernel SVM. (c) Estimated IBM using the domain-invariant kernel SVM. White regions represent 1s and black 0s.

$100,000$ pairs of constraints are used in the kernel learning. We set the distance thresholds u and l to 5% and 95% percentile of the distribution of the observed distances between pairs of points respectively. The slack variable γ and the variance of the Gaussian kernel σ are tuned using cross validation. After we learn the kernel, we train the SVM using another 30 utterances mixed with the same 5 noises. According to Eq. (9), we compute the kernel for these data for SVM training.

To test the system, we use 10 utterances mixed with 12 types of noise—N1: white noise, N2: cocktail party noise, N3: rock music, N4: telephone, N5: fan noise, N6: clock alarm, N7: traffic noise, N8: crowd noise with clap, N9: bird chirping with water flowing, N10: wind noise, N11: rain noise, N12: babble noise. The test noises cover both stationary and non-stationary noises and have very different frequency characteristics. None of the utterances and the noises are seen in the kernel learning and SVM training phase.

As an example, Fig. 1 illustrates mask estimation results for an utterance mixed with an unseen crowd noise with clap at -5 dB using the SVM with the Gaussian kernel and the SVM with the learned domain-invariant kernel respectively. It is clear that the Gaussian kernel SVM leads to severe classification errors because the noise is significantly different from those in the training set. By using kernel learning the system yields a substantially better mask due to the robustness of the learned kernel against different noise types.

To systematically quantify the performance of our system, we compute the HIT rate, defined as the percent of the target-dominant units in the IBM correctly classified, and the false alarm (FA) rate, defined as the percent of the interference-dominant units in the IBM wrongly classified. The difference HIT−FA has been shown to be well correlated to human speech intelligibility [6].

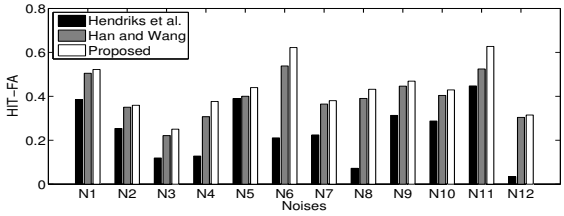


Fig. 2: HIT−FA comparison under unseen noise conditions

Table 1 shows the average classification accuracy and the HIT−FA rates over all 12 noises. We compare the Gaussian kernel SVM (G-SVM) and the SVM with learned domain-invariant kernel (KL-SVM). In the left two columns of the table, in order to eliminate the impact of pitch errors we use ground-truth pitch extracted from the premixed speech [22] to generate the pitch-based features. In the right two columns, we use a pitch estimator [23] to extract pitch from mixtures. Both experiments clearly show that learning the domain-invariant kernel significantly boosts the classification accuracy and the HIT−FA rates under new noise conditions.

Table 1: Average classification accuracy and HIT−FA rates.

	Ground-truth Pitch		Estimated Pitch	
	G-SVM	KL-SVM	G-SVM	KL-SVM
Accuracy	0.742	0.794	0.703	0.746
HIT−FA	0.469	0.537	0.390	0.456

We further compare the proposed method with two other speech separation approaches. The first one is a state-of-the-art speech enhancement algorithm based on a minimum mean-squared error (MMSE) estimator proposed by Hendriks *et al.* [24]. The second one is our previous approach which uses the rethresholding technique to adapt the SVM classification under different noise conditions [14]. The proposed approach in this comparison uses the estimated pitch. As shown in Fig. 2, the proposed approach achieves the highest HIT−FA under every noise condition. On average, the proposed approach outperforms Hendriks *et al.* by 14 percentage points and our previous system by 4 percentage points. We point out that, our previous system needs noise information extracted from the test mixture to adapt the trained model, while the proposed approach can be directly applied to the test mixture and does not need to collect information from the new noise, which is a considerable advantage.

6. CONCLUSION

In this study, we have proposed to learn a domain-invariant kernel to encode speech-related information that is robust to different noise types. With the learned kernel, the speech separation system can be applied to new noise conditions without any prior information of the noise.

7. REFERENCES

[1] D. L. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Speech separation by humans and machines*, P. Divenyi, Ed., pp. 181–197. Kluwer Academic Pub., 2005.

[2] D. S. Brungart, P. S. Chang, B. D. Simpson, and D. L. Wang, "Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation," *J. Acoust. Soc. Am.*, vol. 120, no. 6, pp. 4007–4018, 2006.

[3] N. Li and P. C. Loizou, "Factors influencing intelligibility of ideal binary-masked speech: Implications for noise reduction," *J. Acoust. Soc. Am.*, vol. 123, no. 3, pp. 1673–1682, 2008.

[4] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. of ICML*, 2007, pp. 209–216.

[5] Z. Jin and D. L. Wang, "A supervised learning approach to monaural segregation of reverberant Speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 4, pp. 625–638, 2009.

[6] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *J. Acoust. Soc. Am.*, vol. 126, pp. 1486–1494, 2009.

[7] K. Han and D. L. Wang, "A classification based approach to speech segregation," *J. Acoust. Soc. Am.*, vol. 132, no. 5, pp. 3475–3483, 2012.

[8] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N.D. Lawrence, *Dataset shift in machine learning*, The MIT Press, 2009.

[9] H. Daumé, "Frustratingly easy domain adaptation," in *Proc. of ACL*, 2007, vol. 45, pp. 256–263.

[10] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. of ECCV*, 2010, pp. 213–226.

[11] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms," in *Proc. of IEEE CVPR*, 2011, pp. 1785–1792.

[12] L. Duan, D. Xu, I. W. H. Tsang, and J. Luo, "Visual event recognition in videos by learning from web data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1667–1680, 2012.

[13] A. Ozerov, P. Philippe, F. Bimbot, and R. Gribonval, "Adaptation of bayesian models for single-channel source separation and its application to voice/music separation in popular songs," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 5, pp. 1564–1578, 2007.

[14] K. Han and D. L. Wang, "Towards generalizing classification based speech separation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 21, no. 1, pp. 166–175, 2013.

[15] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: transfer learning from unlabeled data," in *Proc. of ICML*, 2007, pp. 759–766.

[16] D. L. Wang and G. J. Brown, Eds., *Computational auditory scene analysis: Principles, algorithms and applications*, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2006.

[17] Y. Wang, K. Han, and D. L. Wang, "Exploring monaural features for classification-based speech segregation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 2, pp. 270–279, 2012.

[18] V. N. Vapnik, *The nature of statistical learning theory*, Springer, Inc., New York, NY, USA, 2000.

[19] P. Jain, B. Kulis, J. V. Davis, and I. S. Dhillon, "Metric and kernel learning using a linear transformation," *J. Mach. Learn. Res.*, vol. 13, pp. 519–547, 2012.

[20] E. H. Rothauser, W. D. Chapman, N. Guttman, K. S. Nordby, H. R. Silbiger, G. E. Urbanek, and M. Weinstock, "IEEE recommended practice for speech quality measurements," *IEEE Trans. Audio Electroacoustics*, vol. 17, pp. 227–246, 1969.

[21] G. Hu, "100 nonspeech sounds, 2006," http://www.cse.ohio-state.edu/pnl/corpus/HuCorpus.html.

[22] P. Boersma and D. Weenink, *PRAAT: Doing Phonetics by Computer (version 4.5)*, 2007, http://www.fon.hum.uva.nl/praat.

[23] S. Gonzalez and M. Brookes, "A pitch estimation filter robust to high levels of noise (PEFAC)," in *Proc. EUSIPCO*, 2011.

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