



Learning a Speech Manifold for Signal Subspace Speech Denoising

Colin Vaz and Shrikanth Narayanan

Signal Analysis and Interpretation Lab
 University of Southern California, Los Angeles, CA 90089
 cvaz@usc.edu, shri@sipi.usc.edu

Abstract

We present a method for learning a low-dimensional manifold for speech from clean speech samples in high-dimensional space. Using this manifold, we perform speech denoising by projecting noisy speech onto the manifold to remove non-speech components. This method of denoising classifies our algorithm as a signal subspace denoising method, where high-dimensional noisy data is projected onto the signal subspace to recover the signal of interest. We ran denoising experiments with different types of additive noise. The proposed method not only recovers the second formant more accurately, but also produces denoised speech with higher quality (as illustrated by PESQ scores) compared to other signal subspace denoising algorithms.

Index Terms: manifold learning, NPE, manifold charting, speech denoising, signal subspace

1. Introduction

Technological applications using speech are ubiquitous, and include speech-to-text systems [1], emotional-state detection [2], and assistive applications, such as hearing aids [3]. The presence of background noise usually degrades the performance of these systems, thus limiting their use to confined environments or scenarios. Researchers are actively developing speech denoising methods to overcome these barriers. Such methods include signal subspace approaches [4], model-based methods [5], and spectral subtraction algorithms [6]. These different techniques make specific assumptions about the noise or SNR levels, and give a certain trade-off between noise suppression and speech distortion.

Spectral subtraction algorithms are widely used for noise suppression, owing in part to their low computational complexity. A major drawback of these algorithms is the distortion of the speech and introduction of “musical noise”. Furthermore, they require a good estimate of the noise, which is difficult in non-stationary noise conditions. Several researchers have proposed signal subspace denoising methods to overcome the drawbacks of spectral subtraction [4]. The assumption in these approaches is that the speech resides on a low-dimensional subspace of some high-dimensional space. The goal of signal subspace algorithms is to decompose the high-dimensional space into a signal subspace (which predominantly contains speech) and a noise-only subspace. Then, projecting the noisy signal onto the signal subspace should remove many of the noise components and effectively denoise the signal.

Dendrinis et al. proposed the use of the singular value decomposition (SVD) for signal subspace speech denoising in [4]. They calculated the singular values and associated eigenvectors

of the data matrix and projected the noisy data onto the subspace formed by the eigenvectors corresponding to the d largest singular values. Importantly, they described an approach for estimating d assuming the speech is corrupted by white noise. Jensen et al. extended this framework and developed the Truncated Quotient SVD algorithm to denoise speech in colored broadband noise [7]. Their algorithm incorporated a pre-whitening operation into the signal decomposition step. Ephraim and Van Trees used the Karhunen-Loève Transform (KLT) instead of the SVD for signal decomposition [8]. Within the KLT framework, they derived time-domain constrained and spectral-domain constrained estimators to minimize the signal distortion for a certain amount of residual noise. They derived these estimators with the assumption that the noise is white. Hu and Loizou extended this idea to colored noise by building in a pre-whitening step into the decomposition process [9].

Signal subspace methods assume a linear transformation between the high-dimensional space and the signal subspace. However, it is likely that speech lies on a low-dimensional manifold. A manifold is a non-linear space that is approximately linear in small neighborhoods. If the speech indeed lies on a manifold, then this non-linear subspace cannot be properly described by a linear projection. In this paper, we use spectral properties of speech to learn a manifold for speech and its associated non-linear mapping from high-dimensional space to the manifold. We then perform speech denoising by projecting noisy speech onto and manifold, and then projecting the manifold images back into high-dimensional space for signal reconstruction. Since our manifold captures the non-linear geometry of the speech spectrum, we expect our denoising algorithm to generate denoised speech with higher speech quality and lower distortion than other signal subspace denoising methods.

This paper is organized as follows. Section 2 describes the process we used to construct the speech manifold and determine the mappings between the high-dimensional space and the manifold. Section 3 discusses the speech denoising experiments and compares the performance of our algorithm to other signal subspace denoising methods. Section 4 gives insight into the results of our experiments and points out some limitations in our work. Finally, Section 5 offers our conclusions and directions for future work.

2. Manifold Learning Algorithm

We model noisy speech as

$$y[n] = x[n] + \alpha v[n], \quad (1)$$

where $x[n]$ is the clean speech that is corrupted by noise $v[n]$. The α parameter controls the resulting SNR of the noisy speech. We convert the time-domain representation of $y[n]$ to the spectral domain using the short-time Fourier Transform (STFT) with

Work supported by NIH Grant DC007124.

a 25-ms Hamming window shifted by 10 ms. We used a 512-point FFT for the STFT, so the resulting spectral representation is in $D = 257$ -dimensional space (since we are taking the FFT of a real signal, only the first half of the FFT needs to be represented in order to recover the time-domain signal from the FFT). We assume that speech resides on a manifold \mathcal{M} of dimension d within \mathbb{R}^D , with $d < D$. A manifold is a non-Euclidean space that can be approximated by Euclidean patches in small neighborhoods. For example, the surface of the Earth can be thought of as a two-dimensional spherical manifold in three-dimensional Euclidean space, but in small neighborhoods, say at the city level, the Earth is approximately flat (Euclidean). Hence, the Earth can be thought of as a collection of city-sized Euclidean patches joined together to form a sphere. Similarly, one can think of the different speech sounds (phonemes) in speech as Euclidean patches that are joined together to form a speech manifold. Notice that while speech resides on the d -dimensional manifold, it does not confine the noise to the remaining $D - d$ dimensions; rather, we assume that speech does not exist in the remaining $D - d$ dimensions.

To determine the Euclidean phoneme patches, we extracted speech frames from the USC-TIMIT corpus [10] (an articulatory corpus; see Section 3) for each phoneme and obtained spectral representations of the speech frames using STFT. For each phoneme, we embedded its speech frames into a low-dimensional space using Neighborhood Preserving Embedding (NPE) [11]. Unlike methods such as Principal Component Analysis (PCA), which try to preserve the global Euclidean structure of the speech frames when embedding into low-dimensional space, NPE preserves local Euclidean structure. It achieves this by representing \mathbf{x}_{ip} , the i th speech frame labeled as phoneme p , as a linear combination of its K nearest neighbors, and preserving the weights in the linear combination when each speech frame is mapped to a low-dimensional space. This ensures that the local geometry in high-dimensional space will be retained in low-dimensional space. Mathematically, NPE first finds weights $W_{ip,jp}$ such that

$$\sum_{i \in \text{phn}(p)} \|\mathbf{x}_{ip} - \sum_{j \in \text{phn}(p)} W_{ip,jp} \mathbf{x}_{jp}\|_2^2 \quad (2)$$

is minimized, where $W_{ip,jp}$ is the weight of \mathbf{x}_{jp} in the linear combination for \mathbf{x}_{ip} ($W_{ip,jp} = 0$ if \mathbf{x}_{jp} is not a K nearest neighbor of \mathbf{x}_{ip}) and $\text{phn}(p)$ is the set of frame indices labeled as phoneme p . Then, with $W_{ip,jp}$ fixed, it tries to determine \mathbf{y}_{ip} , the low-dimensional embedding of \mathbf{x}_{ip} , by minimizing

$$\sum_{i \in \text{phn}(p)} \|\mathbf{y}_{ip} - \sum_{j \in \text{phn}(p)} W_{ip,jp} \mathbf{y}_{jp}\|_2^2. \quad (3)$$

From this, NPE is able to calculate a matrix $A_p \in \mathbb{R}^{d \times D}$ such that $\mathbf{y}_{ip} = A_p \mathbf{x}_{ip}$. Thus, we are able to find linear mappings A_p for each phoneme from \mathbb{R}^D to \mathbb{R}^d .

Now that we have linear mappings to Euclidean patches for the different phonemes, we need to join these patches together to form a manifold for speech. We use the Manifold Charting method to join these patches together [12]. Manifold Charting determines the best global Cartesian coordinate system to represent the speech frames by re-aligning the coordinate system of each of the phoneme patches. The re-alignment needs to reduce the amount of ‘‘disagreement’’ between the phoneme mappings A_p so that the mapping onto the global coordinate system does its best to respect the local geometry of each phoneme patch. The mapping onto the global coordinate system is found

by minimizing

$$\sum_i \sum_p \sum_q r_{ip} r_{iq} \|\mathbf{y}_{ip} - \mathbf{y}_{iq}\|_2^2, \quad (4)$$

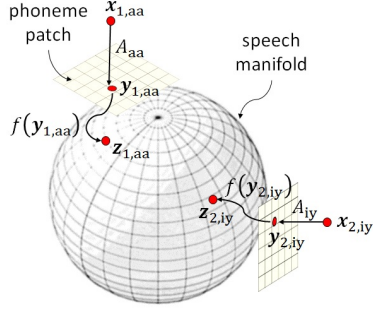
where r_{ip} , called the responsibility, is the probability of the i th speech frame being labeled as phoneme p . This cost function encourages phoneme maps that have a high responsibility for a speech frame to agree on the global coordinate of this speech frame. The solution to the minimization gives a non-linear relationship f between \mathbf{x}_{ip} and the final embedding on the global coordinate system \mathbf{z}_{ip} . Thus, with the phoneme maps A_p from NPE and the global coordinate map f from Manifold Charting, we have a non-linear mapping between the high-dimensional spectral space and the low-dimensional speech manifold: $\mathbf{z}_{ip} = f(\mathbf{y}_{ip}) = f(A_p \mathbf{x}_{ip})$. Figure 1a illustrates the learning procedure.

To perform denoising, we subtract the mean of the estimated noise $\bar{\mathbf{v}}$ from a noisy speech sample $\mathbf{x}_{\text{noisy}}$ to get $\bar{\mathbf{x}}_{\text{noisy}} = \mathbf{x}_{\text{noisy}} - \bar{\mathbf{v}}$. We obtain the noise estimate from noise-only portions of the noisy speech. We then project $\bar{\mathbf{x}}_{\text{noisy}}$ onto the speech manifold using $f(A_p \bar{\mathbf{x}}_{\text{noisy}})$ for each phoneme map p . This gives us the embedding on the manifold $\mathbf{z}_{\text{noisy}}$ and the responsibility that each map has for the noisy sample. We find the phoneme map p_0 that has the highest responsibility, and we use this phoneme map to project the noisy sample $\mathbf{x}_{\text{noisy}}$ onto the manifold and then back into high-dimensional spectral space to get an estimate of the denoised speech sample: $\hat{\mathbf{x}} = A_{p_0}^+ A_{p_0} \mathbf{x}_{\text{noisy}}$, where $A_{p_0}^+$ is the pseudo-inverse of A_{p_0} . Figure 1b gives an example of the denoising procedure. We reconstruct the time-domain signal from $\hat{\mathbf{x}}$ by taking the inverse-FFT using the phase of the noisy signal and using the overlap-add method to combine the time-domain reconstructions from each STFT frame.

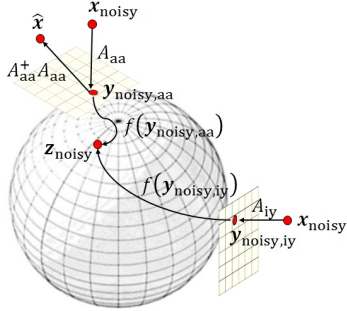
3. Speech Denoising Experiment

We performed a speech denoising experiment using the Electromagnetic Articulatory (EMA) database in the USC-TIMIT corpus [10]. We used the EMA database because it contains clean speech and provides phonetic-level alignments for each utterance. The database consists of 2 male and 2 female native American English speakers speaking 460 TIMIT sentences, and these sentences have 40 unique phonemes. We use the first 300 sentences (training set) from each speaker to learn speaker-dependent speech manifolds as described in Section 2. TIMIT sentences are good for building the manifold because they contain all 40 phonemes in different word-level contexts.

We added white, pink, speech babble, and factory noises from the NOISEX database [13] at 5 dB and 10 dB SNR to the remaining 160 TIMIT sentences. We chose these noises because they are a good mix of wideband, narrowband, stationary, and non-stationary noises. We used noisy sentences 301 to 350 (development set) for determining the optimal dimension d of the manifold and the number of nearest neighbors K in the NPE algorithm. We tuned these parameters to optimize the Perceptual Evaluation of Speech Quality (PESQ) score [14] and the mean square error (MSE) between the denoised speech and the clean speech (reconstruction MSE). The PESQ score provides an automated assessment of speech quality that gives a score for the denoised signal between -0.5 and 4.5 , where -0.5 indicates poor speech quality and 4.5 indicates excellent quality. We determined the optimal parameters to be $d = 40$ and K to be 40% of the number of training samples for each phoneme. The number of samples for each phoneme varies; short-duration



(a) Example of learning the manifold, where speech frame 1 is the aa phoneme and speech frame 2 is the iy phoneme.



(b) Using the manifold for denoising.

Figure 1: Illustration of the manifold learning (a) and denoising (b) procedures. During learning, NPE finds mappings A_{aa} and A_{iy} that maps $\mathbf{x}_{1,aa}$ and $\mathbf{x}_{2,iy}$ on their respective phoneme patches. Manifold charting determines the global coordinates for $\mathbf{y}_{1,aa}$ and $\mathbf{y}_{2,iy}$. During denoising, the algorithm picks the phoneme map, in this case A_{aa} , that had the highest responsibility for generating \mathbf{z}_{noisy} . Then $\hat{\mathbf{x}} = A_{aa}^+ A_{aa} \mathbf{x}_{noisy}$.

phonemes such as stops have fewer samples than long-duration phonemes such as vowels. Thus, we decided to determine K as a proportion of the number of samples per phoneme.

We denoised sentences 351 to 460 (test set) using the manifold we learned. We compared our denoising performance to the signal subspace time-domain constraint (TDC) and spectral-domain constraint (SDC2) algorithms proposed in [9] and the Truncated Quotient SVD (TQSVD) algorithm proposed in [7]. For the TDC and SDC2 algorithms, we used parameter settings that were suggested in [9] and we found these settings to perform well on our data. For the TQSVD algorithm, we tuned the parameters on the development set, optimizing the PESQ score and reconstruction MSE. We evaluated the denoising performance using the PESQ score, the reconstruction MSE, and the log-likelihood ratio (LLR). The LLR gives a measure of the distortion in the spectral envelope caused by the denoising algorithm, and an LLR of 0 indicates no spectral envelope distortion. Table 1 shows these results averaged across the four speakers. Additionally, we used Praat [15] to extract the first three formants (f_1 to f_3) from vowel regions. We found the MSE between the formants extracted from the denoised speech and the formants extracted from the clean speech. This gives us a sense of how well each algorithm preserves information in the spectrum. Table 2 shows the formant MSE results for the male speakers and Table 3 shows these for the female speakers with

the different algorithms in the different noise conditions.

Table 1: PESQ scores, LLR, and reconstruction MSE for all speakers with the different denoising algorithms in different noise conditions.

Metric	SNR (dB)	Noise	Algorithm			
			Manifold	TDC	SDC2	TQSVD
PESQ	10	white	2.72	2.66	2.64	2.39
		pink	2.81	2.64	2.60	2.60
		babble	2.76	2.59	2.55	2.63
		factory	2.77	2.74	2.70	2.59
	5	white	2.49	2.58	2.57	2.20
		pink	2.59	2.50	2.49	2.41
		babble	2.55	2.45	2.44	2.44
		factory	2.55	2.58	2.56	2.40
LLR ($\times 10^{-1}$)	10	white	10.96	5.24	5.27	6.24
		pink	11.00	2.15	2.22	3.81
		babble	10.94	1.74	1.80	3.77
		factory	10.92	1.89	1.97	3.55
	5	white	10.54	1.77	1.74	5.76
		pink	10.95	2.40	2.37	4.03
		babble	11.05	1.93	1.87	3.69
		factory	10.82	1.87	1.86	3.52
Reconstruction MSE ($\times 10^{-4}$)	10	white	4.40	8.99	9.77	9.80
		pink	9.32	13.11	14.39	13.84
		babble	9.82	16.35	18.43	17.32
		factory	9.51	11.34	12.73	15.72
	5	white	8.51	13.40	13.83	13.12
		pink	23.60	22.00	22.44	24.61
		babble	24.88	24.39	25.27	32.42
		factory	24.27	17.48	18.10	29.86

We used the non-parametric Wilcoxon rank-sum test to determine whether the medians of the results are significantly different between the denoising algorithms. For each metric and noise condition, we indicate in the tables the best performing algorithm in bold. Additionally, we indicate in red the best significantly performing algorithm at the 95% level.

4. Discussion

In Table 1, we see that our proposed method produced the best speech quality and lowest reconstruction error compared to the other algorithms in most noise conditions. Notably, our algorithm performed well in white and babble noises. White noise, by its definition, exists in all dimensions of \mathbb{R}^D , so certain components of white noise will reside on the speech manifold, making it impossible to fully remove white noise with our method without additional processing. Speech babble will also have strong components on the manifold. Unfortunately, our algorithm has a high LLR. This likely occurs when the phoneme map responsible for the projection to and from the manifold changes from one frame to the next, resulting in a sudden change in spectral content. While these changes are generally subtle and don't seem to impact the speech quality, it can impact the LLR, which is sensitive to the spectral envelope.

In Tables 2 and 3, we see that our algorithm has a significantly lower error in recovering the second formant for male speakers. While our algorithm also recovered the second formant in female speech well, it was not significantly better than the other algorithms. It is typically harder to recover the second and third formants from noisy speech because they have less energy than the first formant. Thus, the ability of our algorithm to recover the second formant well can benefit applications such as automatic speech recognition (ASR) and speech analysis.

Table 2: Formant MSE for the male speakers with the different denoising algorithms in different noise conditions.

Metric	SNR		Algorithm			
	(dB)	Noise	Manifold	TDC	SDC2	TQSVD
f_1 MSE ($\times 10^4$)	10	white	2.54	2.24	2.34	2.44
		pink	2.25	2.50	2.64	2.00
		babble	2.51	2.60	2.54	2.08
		factory	2.28	2.19	2.35	2.09
	5	white	2.81	2.33	2.43	3.48
		pink	2.32	2.72	2.69	2.59
		babble	2.45	2.58	2.54	2.28
		factory	2.70	2.40	2.56	2.67
f_2 MSE ($\times 10^4$)	10	white	4.22	7.11	7.37	5.12
		pink	4.24	6.60	7.14	5.25
		babble	4.25	5.41	5.75	5.17
		factory	4.19	4.98	5.24	5.51
	5	white	5.61	8.18	7.93	7.09
		pink	5.01	8.42	8.75	6.33
		babble	5.58	6.56	6.74	6.88
		factory	5.68	6.02	6.35	7.75
f_3 MSE ($\times 10^5$)	10	white	1.32	1.40	1.38	1.37
		pink	1.26	1.09	1.11	1.22
		babble	1.19	0.83	0.81	1.39
		factory	1.23	0.93	0.96	1.28
	5	white	1.53	1.72	1.76	1.78
		pink	1.40	1.20	1.23	1.45
		babble	1.41	0.95	0.94	1.68
		factory	1.45	1.17	1.16	1.52

Unlike most signal subspace denoising algorithms, the dimension of our signal subspace (speech manifold) does not change based on the characteristics of the noise. This is because we are trying to explicitly model speech, whereas other signal subspace methods are trying to find the best subspace to project the noisy speech based on some criteria related to the noise. These methods have the advantage of adapting to the acoustic conditions, but are susceptible to breakdown, particularly in low SNR cases, when the assumptions about the noise are not met. Our method is affected less by noise conditions because it does not consider noise when learning the manifold, but it requires more in-depth modeling so the manifold captures the intricacies of speech.

In this paper, we built speaker-dependent speech manifolds. However, we will experiment with building speaker-independent manifolds by incorporating techniques such as spectral normalization to factor out speaker-dependent characteristics. Speaker independence will enable us to denoise speech from speakers for whom we don't have trained a manifold. Moreover, the speaker-independent manifold learning technique can be applied to cases beyond denoising. A good example is acoustic modeling for ASR systems, where the same set of phonemes can be pronounced in different ways and in different acoustic environments. For an ASR to function well, it needs to map all the different acoustic realizations of a particular phoneme to the same phoneme. A manifold learned from multiple speakers will capture the acoustic variability of each phoneme and thus provide the language model in an ASR a consistent phonetic representation of the speech. While acoustic model adaptation is frequently employed to improve ASR performance, we will investigate mapping acoustics onto the speech manifold and performing acoustic modeling on the manifold as a way of building robustness into the acoustic model.

A limitation of our algorithm is the failure to capture temporal aspects of speech when learning the manifold. Using the

Table 3: Formant MSE for the female speakers with the different denoising algorithms in different noise conditions.

Metric	SNR		Algorithm			
	(dB)	Noise	Manifold	TDC	SDC2	TQSVD
f_1 MSE ($\times 10^4$)	10	white	4.29	3.62	3.48	4.01
		pink	4.01	3.78	3.83	3.09
		babble	3.89	3.77	3.84	3.48
		factory	4.10	3.40	3.65	3.26
	5	white	4.93	3.75	3.84	4.86
		pink	4.27	3.87	4.00	3.73
		babble	4.58	4.13	4.04	4.07
		factory	4.57	3.85	3.97	4.08
f_2 MSE ($\times 10^5$)	10	white	1.60	1.66	1.66	1.66
		pink	1.48	1.60	1.67	1.68
		babble	1.56	1.41	1.40	1.70
		factory	1.64	1.44	1.54	1.74
	5	white	1.68	1.72	1.75	1.69
		pink	1.62	1.67	1.67	1.76
		babble	1.70	1.54	1.57	1.83
		factory	1.72	1.48	1.52	2.06
f_3 MSE ($\times 10^5$)	10	white	2.83	2.23	2.28	2.32
		pink	2.69	2.00	2.06	2.32
		babble	2.61	1.47	1.57	2.51
		factory	2.75	1.62	1.67	2.39
	5	white	2.77	2.49	2.52	2.60
		pink	2.84	2.08	2.13	2.46
		babble	2.76	1.68	1.67	2.57
		factory	2.85	2.03	1.98	2.54

phoneme transition patterns can greatly improve distinction between speech and noise. A major reason for the lack of using temporal information when learning the manifold is that current manifold learning formulations do not consider the relationship between points in the high-dimensional space besides spatial relationships. While this may be fine in areas such as image processing, where manifold learning techniques are frequently used [16, 17], this is not adequate for speech data, where temporal connections between speech frames carry a lot of information. One way to overcome this is to develop cost functions in the learning algorithm that consider the spectral and temporal relationships between phonemes, which will enable the manifold to reject acoustics that do not exhibit speech dynamics.

5. Conclusion

We have described a method for learning a speech manifold from clean speech data. Using this manifold, we performed speech denoising akin to signal subspace denoising methods by projecting noisy samples onto the manifold and then projecting them back into high-dimensional space for reconstruction. Quantitative results showed that our proposed algorithm produced the best speech quality with lower reconstruction errors than other signal subspace denoising methods in many of the noise conditions. Moreover, our algorithm recovered the second formant better than the other denoising methods.

We will build on this work by incorporating temporal properties of speech when learning the manifold. We will evaluate our approach on other tasks, such as phoneme classification and acoustic modeling for ASR systems. Moreover, we will develop our method with the goal of providing new insights and visualizations of speech.

6. References

- [1] S. Furui, T. Kikuchi, Y. Shinnaka, and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," *IEEE Trans. Speech and Audio Processing*, vol. 12, no. 4, pp. 401–408, 2004.
- [2] C. M. Lee and S. Narayanan, "Towards detecting emotion in spoken dialogs," *IEEE Trans. Speech and Audio Processing*, vol. 13, no. 2, pp. 293–302, 2005.
- [3] H. W. Löllmann and P. Vary, "Low delay noise reduction and dereverberation for hearing aids," *EURASIP J. Advances in Signal Processing*, vol. 2009, no. 1, 2009.
- [4] M. Dendrinos, S. Bakamidis, and G. Carayannis, "Speech enhancement from noise: A regenerative approach," *Speech Communication*, vol. 10, no. 1, pp. 45–57, 1991.
- [5] Y. Ephraim and D. Mallah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 33, no. 2.
- [6] S. Boll, "Suppression of Acoustic Noise in Speech Using Spectral Subtraction," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.
- [7] S. H. Jensen, P. C. Hansen, S. D. Hansen, and J. A. Srensen, "Reduction of broad-band noise in speech by truncated QSVD," *IEEE Trans. Speech and Audio Processing*, vol. 3, no. 6, pp. 439–448, 1995.
- [8] Y. Ephraim and H. L. V. Trees, "A signal subspace approach for speech enhancement," *IEEE Trans. Speech and Audio Processing*, vol. 3, no. 4, pp. 251–266, 1995.
- [9] Y. Hu and P. Loizou, "A generalized subspace approach for enhancing speech corrupted by colored noise," *IEEE Trans. Speech and Audio Processing*, vol. 11, no. 4, pp. 334–341, 2003.
- [10] S. S. Narayanan, A. Toutios, V. Ramanarayanan, A. Lammert, J. Kim, S. Lee, K. S. Nayak, Y.-C. Kim, Y. Zhu, L. Goldstein, D. Byrd, E. Bresch, P. K. Ghosh, A. Katsamanis, and M. I. Proctor, "Real-time magnetic resonance imaging and electromagnetic articulography database for speech production research (TC)," *J. Acoustical Society of America*, vol. 136, no. 3, pp. 1307–1311, 2014.
- [11] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *Tenth IEEE Int. Conf. Computer Vision*, vol. 2, Oct 2005, pp. 1208–1213.
- [12] M. Brand, "Charting a Manifold," in *Advances in Neural Information Processing Systems*, 2003, pp. 961–968.
- [13] 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.
- [14] Y. Hu and P. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. Audio, Speech, and Language Process.*, vol. 16, no. 1, pp. 229–238, 2008.
- [15] P. Boersma, "Praat, a system for doing phonetics by computer," *Glott International*, vol. 5, no. 9, pp. 341–345, 2001.
- [16] C. Bregler and S. M. Omohundro, "Nonlinear image interpolation using manifold learning," in *Advances in Neural Information Processing Systems*, 1995, pp. 973–980.
- [17] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using Laplacianfaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, March 2005.