Smoothing Speech Trajectories by Regularization

Heyun Huang, Louis ten Bosch, Bert Cranen, Lou Boves

Department of Linguistics, Radboud University Nijmegen,
Erasmaslaan 1, 6525, Nijmegen, the Netherlands
{h.huang,l.tenbosch,b.cranen,l.boves}@let.ru.nl

Abstract

The articulators of human speech might only be able to move slowly, which results in the gradual and continuous change of acoustic speech properties. Nevertheless, the so-called speech continuity is rarely explored to discriminate different phones. To exploit this, this paper investigates a multiple-frame MFCC representation (that is expected to retain sufficient time-continuity information) in combination with a supervised dimensionality reduction method, whose target is to find low-dimensional representations that optimally separates different phone classes. The speech continuity information is integrated into this framework by using the regularization terms that penalize discontinuities. Experimental results on TIMIT phonetic classification show that the use of regularizers can help to improve the separability of phone classes.

Index Terms: Dimensionality Reduction; Contextual Representation; TIMIT Phone Classification; Regularization; Laplacian Smoothing;

1. Introduction

Speech is generated by (semi-)continuous movements of a small number of articulators, each of which are characterized by a small number of degrees of freedom. This suggests that speech signals can be adequately represented by a small number of parameters that vary in a smooth manner as a function of time. Plosives form the single most conspicuous exception to this pattern, because these sounds are characterized by sudden changes in the articulatory system. Still, capturing the articulatory dynamics in the speech signals holds the promise of improving acoustic modeling in automatic speech recognition (ASR) [1–6]. Specifically, capturing trajectories might eliminate the trajectory folding phenomenon in conventional ASR systems [4, 7]. However, despite promising advances in recovering articulatory gestures from the acoustic speech signals (e.g., [8, 9]), purely articulatory-based automatic speech recognition is not yet feasible (and it may never be). Therefore, we need to recur to techniques that allow approximating articulatory dynamics using acoustic representations.

Probably the simplest way for creating representations that represent dynamic articulatory gestures is by concatenating a number of consecutive 10 ms frames. However, such a block of frames captures the underlying gestures only implicitly. If we want to represent syllable-sized pieces of a speech signal, we need approximately 25 frames, for an average syllable duration of 250 ms. If each frame consists of 13 MFCCs (or a similar number of Mel-frequency spectral coefficients) each block comprises over 300 numbers. This is surely a heavily redundant representation, so that there is an obvious need for dimensionality reduction. Ideally, the dimensionality reduction method should help to highlight the dynamic articulatory gestures that have produced the speech signal.

The speech signals that can be generated by continuous movements of articulators that have a only few degrees of freedom will probably be on some low-dimensional manifold in the very high-dimensional acoustic space. Manifold learning [10] has proven to be effective for analyzing trajectory-based signal representations in video processing [11], dynamic texture analysis [12], and speech processing [13]. In [14, 15] it has been shown that manifolds of speech trajectories are effective in separating the overlapping feature spaces occupied by different phones represented by blocks of contiguous frames in phone classification tasks. We believe that this advantage might generalize to acoustical modeling for ASR.

An attractive approach to manifold learning is offered by the graph-embedding supervised dimensionality reduction framework, which is an extension of classical Linear Discriminant Analysis (LDA [16, 17]) (e.g. [18–20]). However, in all approaches to dimensionality reduction attention must be paid to the issue of finding a proper balance between the empirical risk (over-fitting of the high-dimensional data) and the structural risk (failing to meet the requirement that the resulting representations properly reflect the actual degrees of freedom in the physical process that generated the (speech) data). One way in which that balance can be controlled is by imposing additional constraints on the matrix that projects the high-dimensional raw representations into a lower-dimensional space. For example, if we assume that smooth trajectories in articulatory space should result in smooth trajectories of parameters in high-dimensional acoustic space, then one obvious additional requirement is that the weights assigned to corresponding elements of neighboring frames vary smoothly, in order to obtain smooth trajectories in low-dimensional space. From a computational point of view, the most obvious way for imposing additional requirements on the projection matrix is by means of introducing regularization terms, e.g. [13, 21].

In this paper we investigate the application of a series of regularization methods for increasing the smoothness of the projection matrices by their derivatives along the time axis, including the conventional Tikhonov regularizer [14, 21–24], first-order derivative [25], second-order derivative [23, 26] and fourth-order derivative which is also widely adopted as the penalty of smoothness [27].

Phone classification is a multi-class problem, with the number of classes $\boldsymbol{C}$ equal to the number of phone labels that are used in the labeling of a training corpus. Multi-class classification can be approached in three ways: by means

The research of Heyun Huang has received funding from European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement no. 213850. Dr. Louis ten Bosch has received funding from the OPTIFOX project.

of a classifier that uses $\boldsymbol{C}$ models in parallel and assigns an unknown observation to the class c_c that returns the best match; by combining the results of $\boldsymbol{C}$ binary classifiers that separate class c_c, $c = 1, 2, \cdots, C$ from all $C-1$ remaining classes, or by integrating the results of $\frac{C(C-1)}{2}$ binary classifiers that separate all pairs of two classes. In this paper we focus on separating *pairs* of highly confusable classes [14, 21]. We opt for this form of binary classification because the results are useful for phonetic research and advanced acoustic modeling in ASR. Different pairs of classes, even within a broad phonetic class might need different features for their separation. Such a targeted approach is difficult or impossible to implement in a multi-class classification strategy.

The rest of this paper is organized as follows: In Section 2 we briefly introduce various regularizers, and explain how they impose smoothness constraints on the mapping into a lower-dimensional space by the specific variant of LDA that we use in this research, i.e., Locally Discriminant Embedding (**LDE**) [28]. In Section 3 we explain the design of the experiments. Section 4 presents the results of the experiments. Discussion and conclusions are presented in Section 5.

2. Regularization for Speech Trajectories

Our phone classification experiments are based on the TIMIT corpus [29], which comes with accurate labels and segmentation. This allows us to represent all phone tokens by a sequence of $N = 23$ MFCC frames, centered around the middle frame of this phone. Each frame comprises $M = 13$ coefficients. Thus, in a binary classification setting each token is represented as a matrix $\mathbf{X}^c$.

2.1. Supervised Dimensionality Reduction

Supervised dimensionality reduction algorithms first vectorize the matrix representation to $\mathbf{x}_i^c \in \mathbb{R}^D$, $D = 13 \times 23$, and then find the projection matrix $\mathbf{W} \in \mathbb{R}^{D \times d}$ with which a d-dimensional representations $\mathbf{z}_i \in \mathbb{R}^d$ can be obtained by $\mathbf{z}_i = \mathbf{W}^T \mathbf{x}_i$ that maximizes the separation between the two classes. In the traditional LDA approach $\mathbf{W}$ is found by

$$\underset{\mathbf{W}}{\operatorname{argmax}} \left(\frac{tr(\mathbf{W}^T \mathbf{S}^{(b)} \mathbf{W})}{tr(\mathbf{W}^T \mathbf{S}^{(w)} \mathbf{W})} \right) \tag{1}$$

In this paper we replace the traditional LDA by Locally Discriminant Embedding (LDE) approach, which is a form of manifold learning [28]. In LDE the scatter matrices $\mathbf{S}^{(w)}$ and $\mathbf{S}^{(b)}$ are defined as

$$\mathbf{S}^{(w)} = \frac{1}{2} \sum_{ij} A_{ij}^w (x_i - x_j)(x_i - x_j)^T \tag{2}$$

$$A_{ij}^w = \begin{cases} \frac{1}{U_w} & x_i/x_j \text{ is nearest neighbor of } x_j/x_i \\ & x_i \text{ and } x_j \text{ are from the same class} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{S}^{(b)} = \frac{1}{2} \sum_{ij} A_{ij}^b (x_i - x_j)(x_i - x_j)^T \tag{3}$$

$$A_{ij}^b = \begin{cases} \frac{1}{U_b} & x_i/x_j \text{ is nearest neighbor of } x_j/x_i \\ & x_i \text{ and } x_j \text{ are from different classes} \\ 0 & \text{otherwise} \end{cases}$$

in which two parameters U_w and U_b are the numbers of the nearest neighbors from the same class and the other class, respectively. Therefore, the manifold information in the feature space of $\mathbf{x}^c \in \mathbb{R}^D$ is captured by the nearest neighbor graphs.

When conventional LDA is used as a classifier in its own right, only the first $\boldsymbol{C} - 1$ columns in the transformation matrix $\mathbf{W}$ are relevant. However, if LDA is used for dimensionality reduction there is no uniquely defined upper bound on the dimension of the target space. Therefore, we will determine the optimal dimensionality (d) of the target space experimentally.

2.2. Regularization for obtaining smooth trajectories

Eq. (1) is an over-determined system. Without imposing additional constraints on the solution, this conventional solution to $\mathbf{W}$ will favor the empirical risk, at the cost of the structural risk. The weight of the structural risk can be increased by adding a regularization term $\mathbf{R}$, which leads to

$$\underset{\mathbf{W}}{\operatorname{argmax}} \left(\frac{tr(\mathbf{W}^T \mathbf{S}^{(b)} \mathbf{W})}{tr(\mathbf{W}^T [(1-\gamma)\mathbf{S}^{(w)} + \gamma \mathbf{R}] \mathbf{W})} \right) \tag{4}$$

In Eq. (4) γ ($0 \leq \gamma \leq 1$) is the weight given to the regularization term $\mathbf{R}$. Different choices for the matrix $\mathbf{R}$ result in different solutions for minimizing the structural risk.

When we represent phones as a block of 23 frames of 13 MFCC features we have a 299-dimensional space, in which we have no more than a couple of hundred tokens of each class. Thus, we have a small-sample-size problem [23, 24]. This means that the scatter matrices, especially the within-scatter matrix, are poorly estimated. This is likely to make the projection matrix $\mathbf{W}$ difficult to interpret in physical terms. For instance, the inter-correlation of features along time, caused by the continuous movements of the articulators, is probably under-estimated. Moreover, some spurious structures, such as the correlation among MFCCs, which is expected to have been removed by the discrete cosine transform, might re-appear in the scatter matrices. Ideally, to be interpretable, the projection matrix $\mathbf{W}$, which can be considered as a set of basis vectors, should "match" the continuity properties of the input, in our case the smooth speech trajectories represented in $\mathbf{X}$. Therefore, there is a necessity for regularizing the scatter matrix to be consistent with the underlying physical processes by smoothing the projection matrix along the time axis.

In the following we introduce four regularization terms to enhance the smoothness and therewith the interpretability of the projection matrix, by defining $\mathbf{R}$ in four different ways.

2.2.1. Enhancing Interpretability by Tikhonov Regularization

The simplest way for imposing smoothness on the transformation matrix $\mathbf{W}$ is by limiting its variance. This can be accomplished by using Ridge Regression Regularization [22] or Tikhonov Regularization [21, 23, 24] by setting $\mathbf{R} = \mathbf{I}$, which means minimizing the L_2-norm of $\mathbf{W}$.

This regularizer effectively imposes an upper bound on the variance of the coefficients in the transformation matrix $\mathbf{W}$, which is equivalent to biasing the estimator. When setting $\mathbf{R} = \mathbf{I}$ and increasing the value of γ, Eq. (4) will increase the bias in the estimate of the within-class scatter matrix. The bias will weaken the impact of eigenvectors with large eigenvalues, while the impact of eigenvectors with small eigenvalues will be strengthened. However, the Tikhonov regularizer can only decrease the overall variance; it cannot impose local smoothness.

For $\gamma = 0$ Eq. (4) defaults to the unbiased solution. The optimal value of γ for a specific application and data set must be found experimentally in a cross-validation design. This also holds for other choices of $\mathbf{R}$.

2.2.2. Enhancing Continuity by Laplacian Smoothing

While the Tikhonov regularization reduces the overall variance in the transformation matrix, there is no guarantee that the derivatives of the sequence of coefficients along the time dimension are smooth. To realize this, the projection matrix $\mathbf{W}$ must retain the time continuity of the trajectories. "Continuity" can be linked to the smoothness of the discriminant functions that make up $\mathbf{W}$ by minimizing the derivatives of the row vectors (over time). The second-order derivative is the most commonly-used measure for the smoothness [23,26,27], which can be quantified in terms of the coefficients in row m of the $M(=13) \times N(=23)$ matrix $\mathbf{W}$:

$$\begin{aligned} J_2(\mathbf{w}_m) =& (w_{m1} - w_{m2})^2 + \\ &+ \sum_{i=2}^{N-2} (w_{m,i-1} - 2w_{mi} + w_{m,i+1})^2 \\ &+ (w_{m,N-1} - w_{mN})^2 \end{aligned} \quad (5)$$

With this definition, the smoothness of the projection matrix $\mathbf{W}$ is the sum of Eq. 5 over all rows: $J_2(\mathbf{W}) = \sum_m J_2(\mathbf{w}_m)$. This can be reformulated in matrix terms as $J_2(\mathbf{W}) = \mathbf{W}^T \mathbf{D_2} \mathbf{D_2^T} \mathbf{W}$, where the 23×23 matrix $\mathbf{D_2}$ is the so-called Laplacian smoother or Neumann discretizer [23,26]:

$$\mathbf{D_2} = \frac{1}{h_2^2} \begin{pmatrix} -1 & 1 & & & & & & 0 \\ 1 & -2 & 1 & & & & & \\ & 1 & -2 & 1 & & & & \\ & & . & . & . & & & \\ & & & 1 & -2 & 1 & & \\ & & & & 1 & -2 & 1 & \\ 0 & & & & & 1 & -1 & \end{pmatrix} \quad (6)$$

The weight factor $1/h_2^2$ is related to the number of grid points on which the second order derivative along the time dimension is estimated. Substituting $\mathbf{D_2} \mathbf{D_2^T}$ for $\mathbf{R}$ in Eq. 4 yields the so-called Smoothed LDE method using the 2nd-order derivatives (SLDE-2 method). Actually, the first-order and fourth-order derivatives are also widely used for smoothness penalty [25,27]. They are obtained by replacing $\mathbf{D_2}$ by $\mathbf{D_1}$ for the first-order derivative and $\mathbf{D_4}$ for the fourth-order derivative:

$$\mathbf{D_1} = \frac{1}{h_1^2} \begin{pmatrix} -1 & 1 & & & & & & 0 \\ & -1 & 1 & & & & & \\ & & -1 & 1 & & & & \\ & & . & . & . & & & \\ & & & & -1 & 1 & & \\ 0 & & & & & -1 & 1 & \end{pmatrix} \quad (7)$$

$$\mathbf{D_4} = \frac{1}{h_4^2} \begin{pmatrix} 1 & -2 & 1 & & & & & 0 \\ -2 & 5 & -4 & 1 & & & & \\ 1 & -4 & 6 & -4 & 1 & & & \\ & 1 & -4 & 6 & -4 & 1 & & \\ & & . & . & . & & & \\ & & & 1 & -4 & 6 & -4 & 1 \\ & & & & 1 & -4 & 5 & -2 \\ 0 & & & & & 1 & -2 & 1 \end{pmatrix} \quad (8)$$

In fact, the Tikhohov regularizer that penalizes the norm of $\mathbf{W}$ could be viewed as an approach to penalizing the zeroth-order derivative ($\mathbf{D_0} = \mathbf{I}$). Therefore, in this paper, we will investigate four variants of the Smooth LDE (SLDE), named by the order of derivative it is based upon (SLDE-n, $n = 0, 1, 2, 4$). To be more specific, in SLDE-n the matrix $\mathbf{R}$ in Eq. (4) is replaced by $\mathbf{D_n} \mathbf{D_n^T}$.

3. Experimental Setup

3.1. The Data: TIMIT

In our phone classification experiments with TIMIT we adhered to the standard division of the corpus in training, testing, and tuning data [30]. We use the reduced label set proposed in [31]: the original 64 labels are collapsed into 48 phone labels, excluding the glottal stops. We reduce the number of relevant phone classes further, by excluding all forms of 'silence'; this leaves us with 44 phone classes.

A short-time Fourier transform is performed on each utterance with a 25 millisecond Hamming window which is shifted in 10 millisecond steps. The Fourier coefficients are transformed into 13 MFCCs: $c_0 - c_{12}$. Phone tokens are represented by a block of 23 frames, whose center frame is aligned with that of the phone.

3.2. Classification Task

We investigate binary classification with $\frac{44 \times (44-1)}{2} = 946$ pairs of phone classes. Actually, a large number of classes are hardly ever confused. For example, the classification between vowels and voiceless consonants are likely to achieve (nearly) 100% accuracy. Even for some pairs from one broad phonetic class there are virtually no confusions. Therefore, we focused on the class pairs whose feature space overlap substantially; knowing how these pairs can be separated is crucial to understand the phonetic feature space. For this purpose, we define "confusability" as follows: for each phone pair, if any of those compared methods yields a classification accuracy lower than 0.90 or 0.95, that phone pair will be referred to as "highly confusable" or "fairly confusable", respectively.

3.3. Classification Strategy

In our experiments we first reduce the dimensionality of the phone representations, and subsequently use a weighted k Nearest Neighbor (WkNN) classifier for the eventual binary classification task [10]. For any test vector $\mathbf{t}$, we first find its k nearest neighbors in the training set: $\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_k$. The weights of these neighbors are accumulated by $w_i = \exp(-||\mathbf{t}_i - \mathbf{t}||^2/\tau), i = 1, 2, ..., k$, in which τ controls the influence of neighbors. The phone label assigned to $\mathbf{t}$ is the class with the highest aggregated weights.

The LDE procedure for dimensionality reduction requires setting three parameters, viz. size of the between-class neighborhood U_b and the within-class neighborhood U_w, as well as the number of dimensions d of the target space. In [32] it was found that the classification performance is not sensitive to variations in the value of U_b and U_w, as long as these parameters have values in a reasonable range. For the research in this paper we set $U_b = 20$ and $U_w = 6$. We also found that the optimal dimensionality of the target space is $d = 10$ for consonants and $d = 25$ for vowels. The results provided in the following section are based on these LDE settings. The h in the prefactor of $\mathbf{D}$s can be omitted: we normalize $\mathbf{S}^w$ and $\mathbf{R}$ to achieve an interpretable balance between them. The WkNN classifier has two parameters that must be set. In a number of preliminary

Table 1: Statistics of the number of pairs and used tokens of the "fairly confusable" sets (≤ 0.95), for broad phonetic classes: Plosives (PL), Strong Fricatives (SF), Weak Fricatives (WF), Nasals (NS), Semi-Vowels (SeV), Short Vowels (ShV), and Long-Vowels (LoV)

Nums	PL	SF	WF	NS	SeV	ShV	LoV
Pairs	11	4	5	5	4	18	13
Tokens	2841	941	857	1833	1373	6367	2649

Table 2: Statistics of the number of pairs and used tokens of the "highly confusable" sets (≤ 0.90), for broad phonetic classes: Plosives (PL), Strong Fricatives (SF), Weak Fricatives (WF), Nasals (NS), Semi-Vowels (SeV), Short Vowels (ShV), and Long-Vowels (LoV)

Nums	PL	SF	WF	NS	SeV	ShV	LoV
Pairs	2	1	1	1	2	9	3
Tokens	476	502	169	557	669	3149	634

experiments we searched for optimal values of these parameters in the ranges $15 \leq k \leq 40$ and $3.5 \leq \tau \leq 6.5$. It appeared that varying these parameters does not have a significant effect. Therefore, we select $k = 25$ and $\tau = 4.5$ for the configuration of the WkNN classifier in the remainder of this paper.

3.4. Comparing the Methods

We will compare the original method LDE with the four regularized variants. To fully and fairly investigate their effectiveness, we tune the weight γ in Eq. (4) on the following grid: from 0 to 0.1 (stepsize 0.01), from 0.1 to 0.9 (0.05), from 0.9 to 0.99 (0.01), and finally from 0.99 to 0.999 (0.001).

4. Experimental Results

4.1. Classification Per Broad Class

Since different broad phonetic classes probably contains phones produced by disparate mechanisms, this subsection reports the classification results for each broad phonetic class: Plosives (PL), Strong Fricatives (SF), Weak Fricatives (WF), Nasals (NS), Semi-Vowels (SeV), Short Vowels (ShV), and Long-Vowels (LoV). According to Subsection 3.2, only the "fairly confusable" and "highly confusable" binary pairs are concerned. Table 1 and Table 2 introduce the numbers of phone pairs and tokens for each broad phonetic classes. It should be mentioned that the number of tokens might be larger than the number of overall tokens in each broad phonetic class since parts of phones might be visited more than once. For instance, the plosive $/b/$ is easily confused by $/p/$ and $/g/$, and thus the test samples of $/b/$ will be at least used twice.

The details of classification accuracy underlying the data are shown in Table 3 (for "fairly confusable" pairs) and Table 4 (for "highly confusable" pairs). Each number (classification accuracy) in these tables is computed by the number of correctly tokens divided by the number of used tokens (given in Table 1 and Table 2), for each broad class. These numbers can be interpreted by the weighted average of binary classifiers and the weight of a binary classifier depends on the frequency of two classes of phones with which this classifier is involved.

Table 3: Performance comparison of the average binary classification accuracy of five methods: LDE and SLDE-n ($n = 0, 1, 2, 4$) on the fairly confusable pairs (≤ 0.95), for broad phonetic classes: Plosives (PL), Strong Fricatives (SF), Weak Fricatives (WF), Nasals (NS), Semi-Vowels (SeV), Short Vowels (ShV), and Long-Vowels (LoV)

	LDE	SLDE-0	SLDE-1	SLDE-2	SLDE-4
PL	92.90	93.18	93.33	**93.53**	93.33
SF	91.32	90.92	91.59	91.45	**91.60**
WF	91.72	92.51	93.26	**93.28**	93.10
NS	89.69	90.18	90.27	90.37	**90.63**
SeV	91.59	92.35	92.72	**93.10**	92.99
ShV	89.29	89.70	**89.87**	89.73	89.85
LoV	92.46	93.37	**93.66**	93.34	93.53

Table 4: Performance comparison of the average binary classification accuracy of five methods: LDE and SLDE-n ($n = 0, 1, 2, 4$) on the highly confusable pairs (≤ 0.90), for broad phonetic classes: Plosives (PL), Strong Fricatives (SF), Weak Fricatives (WF), Nasals (NS), Semi-Vowels (SeV), Short Vowels (ShV), and Long-Vowels (LoV)

	LDE	SLDE-0	SLDE-1	SLDE-2	SLDE-4
PL	88.13	88.76	89.02	**89.56**	89.36
SF	88.05	87.05	**88.25**	**88.25**	**88.25**
WF	85.80	88.76	**89.92**	89.15	89.66
NS	81.51	82.94	83.30	83.30	**83.75**
SeV	88.34	89.12	89.54	**90.47**	90.06
ShV	85.28	85.75	**85.88**	85.60	85.75
LoV	86.04	**86.91**	86.69	86.67	**86.91**

The most important information from these tables is that nearly all regularization terms seem to be beneficial to the original LDE algorithm. This suggests that the projection matrix $\mathbf{W}$ does require some structures to prevent the arbitrary shapes. The "SLDE-0" method, which actually is the Tikhonov regularizer, achieves the performance slightly inferior to that of other methods penalizing the "$\geq 0''$-order derivatives, especially for those consonants (PL, SF, WF, NS and SeV in Table 3 and Table 4). For vowels (referring to ShV and LoV), the Tikhonov regularizer yields competitive classification accuracy with the other three regularizers. This might imply that the trajectory structures has been fully explored in $\mathbf{S}^w$ for the vowels, but less so for the consonants. Among the remaining three regularization terms, there is no one obviously outperforming the other two in all cases, which means that the manifold of the phones from different broad classes have different shapes and thus should be smoothed in different ways.

4.2. Analysis on the Tuning Parameter γ

An alternative to analyze the effectiveness of regularizers is to explore how the performance varies as a function of the weighing parameter γ. This is shown in Fig. 1 for consonants and Fig. 2 for vowels.

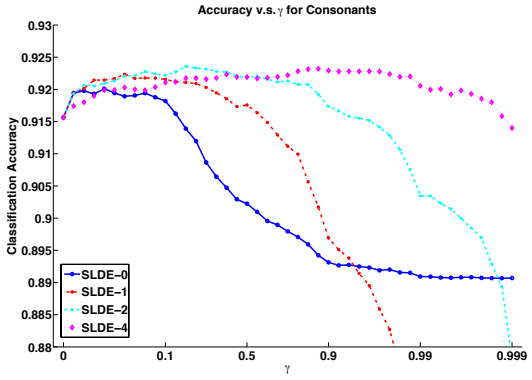

Figure 1: Classification accuracy of "fairly confusable" consonants as a function of the smoothness parameter γ.

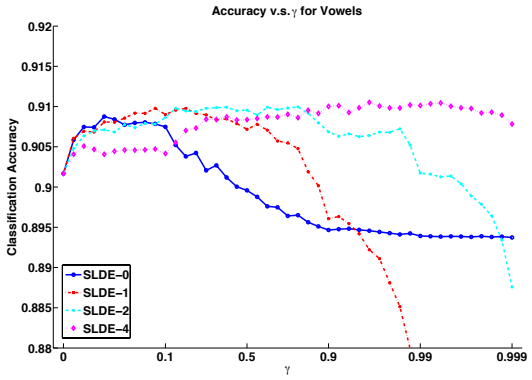

Figure 2: Classification accuracy of "fairly confusable" vowels as a function of the smoothness parameter γ.

Both figures refine our earlier observation that adding the regularization term is beneficial to LDE (LDE corresponds to $\gamma = 0$). Moreover, the performance of "SLDE-0" is comparable with that of other three regularization terms for vowels but not for consonants. This can be deduced from the observation that the peak region of "SLDE-0" (the blue curve in Fig. 2) stays on a similar horizontal level as the other curves do. However, for consonants, the peaks of "SLDE-n" ($n \geq 0$) are consistently higher than those of "SLDE-0" (c.f. Fig. 1). The figures show that the three methods "SLDE-n" ($n > 0$) achieve similar performance.

The three smoothness regularizers have different meaningful γ-ranges, where their performance is superior to the original LDE ($\gamma = 0$). Since all these regularizers adopt the same exhaustive tuning grid and all related matrices ($\mathbf{S}^w$ and different $\mathbf{R}$s) are normalized, we argue that the higher the order of the derivative of $\mathbf{W}$ we use, the most robust the performance is. More precisely, SLDE-4 (corresponding to the magenta curves with "diamonds") outperforms LDE for almost all values of γ, while for $n < 4$ "SLDE-n" has a narrower optimal γ-range.

In summary, this sensitivity analysis further substantiates the effectiveness of introducing $\mathbf{R}$ into LDE and indicates the robustness for variations in γ when applying the fourth-order derivative of $\mathbf{W}$ for regularization.

4.3. The Impact of $\mathbf{R}$ on $(\mathbf{S}^w + \gamma\mathbf{R})^{-1}$

In this part, the deeper analysis on the impact of the regularization term is given by showing the structural alteration of the inverse matrix of the regularized within-class scatter matrix: $(\mathbf{S}^w + \gamma\mathbf{R})^{-1}$. This is due to the fact that $(\mathbf{S}^w + \gamma\mathbf{R})^{-1}\mathbf{S}^b$ directly determines the projection matrix $\mathbf{W}$. Fig. 3 is an example with $\mathbf{R} = \mathbf{D_2}\mathbf{D_2}^T$. We choose $\mathbf{D_2}$ to guarantee the inferior results when γ approaches 0 and 1. Four images in this figure are 23×23, which correspond to the counterpart of $(\mathbf{S}^w + \gamma\mathbf{R})^{-1}$ of the first MFCCs of 23 frames. As indicated in the figure, the upper-left one is the original LDE, meaning $\gamma = 0$. The upper-right and the lower-left images are with the small and medium γs. When γ increases (from "zero" to "small" to "medium"), the corresponding image will have a wider "band", which means the enhancement of the structures over time. Referring to Fig. 1 and Fig. 2, the time-trajectory structure might explain the gain achieved by enlarging γ.

However, the lower-right image, whose γ is large enough to suppress the contribution of $\mathbf{S}^w$ when computing $(\mathbf{S}^w+\gamma\mathbf{R})^{-1}$, almost loses the structure of $(\mathbf{S}^w)^{-1}$, especially on its diagonal part. This might explain the (great) inferior performance when γ approaches 1 in Fig. 1 and Fig. 2.

Therefore, it might be concluded the impact of $\mathbf{R}$ on $\mathbf{S}^w$ is simultaneously enhancing the time-trajectory structure and retaining the details of $\mathbf{S}^w$ with the trade-off parameter γ.

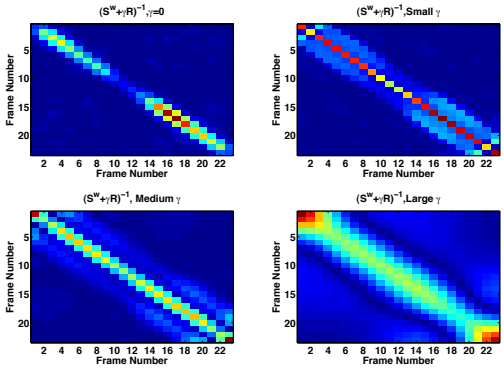

Figure 3: Four special cases of $(\mathbf{S}^w + \gamma\mathbf{R})^{-1}$: $\gamma = 0$, small γ, medium γ, and large γ. The shown part corresponds to 23 frames of first MFCC.

5. General Discussion and Conclusion

To model the time-continuity of speech trajectories for the purpose of phonetic classification, this paper introduces the idea of smoothing the projection matrix of Linear Discriminant Embedding [28], a supervised dimensionality reduction method adopting the manifold information which was proven superior over the conventional Linear Discriminant Analysis. The basic idea is realized by penalizing $\mathbf{W}$'s derivatives to enhance the smoothness of the underlying manifold. Specifically, four orders of derivatives are investigated in this paper, namely the zeroth-order (Tikhonov regularizer), first-order, second-order, and fourth-order derivatives. All of them are implemented by adding the specified regularization term $\mathbf{R}$ to the within-class

scatter matrix $\mathbf{S}^w$.

The binary classification performance on the confusable phone pairs briefly reveals the effectiveness of imposing the regularization terms. When comparing the four alternatives, the zeroth-order one appears to be moderately inferior to the other three comparable regularizers for consonants but yield competitive classification accuracy in the case of vowels. This might indicate a larger necessity of exploiting the continuity within the trajectories for consonants compared to vowels. The robustness of γ in Eq. (4) is also explored. Both Fig. 1 and Fig. 2 indicate that higher-order derivatives can generate a more robust regularizer: using the fourth-order derivative results in improvement over LDE for nearly all γs in the $[0, 1]$-grid. More importantly, the reason why the regularization term $\mathbf{R}$ leads to improvement is likely to be related to how $\mathbf{R}$ influences structures of $(\mathbf{S}^w)^{-1}$ over time (i.e. "frame number" in Fig. 3). The observation is that tuning γ should enhance the time-trajectory structure without losing too much the details in $\mathbf{S}^w$. The analysis on $(\mathbf{S}^w + \gamma\mathbf{R})^{-1}$ suggests that regularization is an effective way to integrate prior information into a small-sample-size problem.

In the near future, our research aim is to analyze the speech production mechanisms, relate them with the smoothness regularizer and thus get more insight into the feature space of speech trajectories. Another interesting topic is to integrate this approach with ASR. Probably regularization is useful for noise-robust ASR since the time-continuity of speech might be robust to environmental noise. Finally, a theoretical analysis on how exactly $\mathbf{R}$ perturbs the eigenvectors of $(\mathbf{S}^w + \gamma\mathbf{R})^{-1}\mathbf{S}^b$ will be a research topic in the near future.

6. References

[1] H.Gish and K.Ng, "Parametric trajectory models for speech recognition," in *Proc. of ICASSP*, 1996, pp. 466–469.

[2] Y. Gong, "Stochastic trajectory modeling and sentence searching for continuous speech recognition," *IEEE Transaction Speech and Audio Processing*, vol. 5, pp. 33–44, January 1997.

[3] M. Lieb and R. Haeb-Umbach, "LDA derived cepstral trajectory filters in adverse environmental conditions," in *Proc. of ICASSP*, 2000, pp. 1105 – 1108.

[4] Y. Han, J. de Veth, and L. Boves, "Trajectory clustering for solving the trajectory folding problem in automatic speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15:4, pp. 1425 – 1434, 2007.

[5] D. Yu, L. Deng, and A. Acero, "Speaker-adaptive learning of resonance targets in a hidden trajectory model of speech coarticulation," *Computer Speech and Language*, vol. 21, no. 4, pp. 72 – 87, 2007.

[6] B. Zhao and T. Schultz, "Toward robust parametric trajectory segmental model for vowel recognition," in *Proc. of ICASSP*, 2002.

[7] I. Illina and Y. Gong, "Elimination of trajectory folding phenomenon: HMM, trajectory mixture HMM and mixture stochastic trajectory model," in *Proc. of ICASSP*, 1997.

[8] P. K. Ghosh and S. Narayanan, "Automatic speech recognition using articulatory features from subject-independent acoustic-to-articulatory inversion," *The Journal of the Acoustical Society of America*, vol. 130, no. 4, pp. EL251–EL257, 2011.

[9] V. Mitra, H. Nam, C. Espy-Wilson, E. Saltzman, and L. Goldstein, "Articulatory information for noise robust speech recognition," *IEEE Transactions On Audio, Speech, And Language Processing*, vol. 19, pp. 1913 – 1924, 2011.

[10] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extension: A general framework for dimensionality reduction," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 40–51, 2007.

[11] A. Rahimi, B. Recht, and T. Darrell, "Learning appearance manifolds from video," in *Proc. of CVPR*, 2005.

[12] Y. Liu, Y. Liu, and K. Chan, "Nonlinear dimensionality reduction with hybrid distance for trajectory representation of dynamic texture," *Signal Processing*, vol. 90, pp. 2375–2395, 2010.

[13] A. Jansen and P. Niyogi, "Intrinsic Fourier analysis on the manifold of speech sounds," in *Proc. of ICASSP*, 2006, pp. 241–244.

[14] P. Clarkson and P. Moreno, "On the use of support vector machines for phonetic classification," in *Proc. of ICASSP*, 1999, pp. 585–588.

[15] T. N. Sainath, A. Carmi, D. Kanevsky, and B. Ramabhadran, "Bayesian compressive sensing for phonetic classification," in *Proc. of ICASSP*, 2010, pp. 4370–4373.

[16] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," in *Proc. of ICASSP*, 1992, pp. 13 – 16.

[17] T. Eisele, R. Haeb-Umbach, and D. Langmann, "A comparative study of linear feature transformation techniques for automatic speech recognition," in *Proc. of ICSLP*, 1996, pp. 252 – 255.

[18] M. Sakai, N. Kitaoka, and K. Takeda, "Feature transformation based on discriminant analysis preserving local structure for speech recognition," in *Proc. of ICASSP*, 2009, pp. 3813 – 3816.

[19] H. Huang, Y. Liu, J. Gemmeke, L. ten Bosch, B. Cranen, and L. Boves, "Globality-locality consistent discriminant analysis for phone classification," in *Proc. of INTERSPEECH*, 2011.

[20] N. Kumar and A. Andreou, "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Communication*, vol. 26, pp. 283–297, 1998.

[21] R. Rifkin, K. Schutte, M. Saad, J. Bouvrie, and J. Glass, "Noise robust phonetic classification with linear regularized least squares and second-order features," in *Proc. of ICASSP*, 2007.

[22] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55 – 67, 1970.

[23] T. Hastie, A. Buja, and R. Tibshirani, "Penalized discriminant analysis," *The Annals of Statistics*, vol. 23, pp. 73 – 102, 1994.

[24] J. Friedman, "Regularized discriminant analysis," *Journal of the American Statistical Association*, pp. 165 – 175, 1989.

[25] W. Zuo, L. Liu, K. Wang, and D. Zhang, "Spatially smooth subspace face recognition using LOG and DOG penalties," in *Advances in Neural Network, ISNN*.

[26] D. Cai, X. He, Y. Hu, J. Han, and T. Huang, "Learning a spatially smooth subspace for face recognition," in *Proc. of IJCAI*, 2007.

[27] J. Ramsay, G. Hooker, and S. Graves, *Functional Data Analysis with R and MATLAB*. Springer, 2005.

[28] H.-T. Chen, H.-W. Chang, and T.-L. Liu, "Local discriminant embedding and its variants," in *Proc. of CVPR*, 2005, pp. 846–853.

[29] L. Lamel, R. Kassel, and S. Seneff, "Speech database development: Design and analysis of the acoustic-phonetic corpus," in *Proc. of DARPA Speech Recognition Workshop*, 1986.

[30] A. Halberstadt, "Heterogeneous acoustic measurements and multiple classifiers for speech recognition," *Ph.D. Thesis, MIT*, 1998.

[31] K. F. Lee and H. W. Hon, "Speaker-independent phone recognition using HMMs," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 11, pp. 1641–1648, 1989.

[32] H. Huang, Y. Liu, and L. Boves, "Investigation of supervised dimensionality reduction methods for phonetic classification," in *Proc. of ACM ICIMCS*, 2011.