Kernel Eigenvoice Speaker Adaptation

Brian Mak, James T. Kwok, and Simon Ho

Abstract

Eigenvoice-based methods have been shown to be effective for fast speaker adaptation when only a small amount of adaptation data, say, less than 10 seconds, is available. At the heart of the method is principal component analysis (PCA) employed to find the most important eigenvoices. In this paper, we postulate that nonlinear PCA using kernel methods may be even more effective. The eigenvoices thus derived will be called *kernel eigenvoices* (KEV), and we will call our new adaptation method *kernel eigenvoice speaker adaptation*. However, unlike the standard eigenvoice (EV) method, an adapted speaker model found by the kernel eigenvoice method resides in the high-dimensional kernel-induced feature space, which, in general, cannot be mapped back to an exact pre-image in the input speaker supervector space. Consequently, it is not clear how to obtain the constituent Gaussians of the adapted model that are needed for the computation of state observation likelihoods during the estimation of eigenvoice weights and subsequent decoding. Our solution is the use of composite kernels in such a way that state observation likelihoods can be computed using only kernel functions without the need of a speaker-adapted model in the input supervector space. In this paper, we investigate two different composite kernels for KEV adaptation: direct sum kernel and tensor product kernel. In an evaluation on the TIDIGITS task, it is found that KEV speaker adaptation using both forms of composite Gaussian kernels are equally effective, and they outperform a speaker-independent model and adapted models found by EV, MAP, or MLLR adaptation using 2.1s and 4.1s of speech. For example, with 2.1s of adaptation data, KEV adaptation outperforms the speaker-independent model by 27.5%, whereas EV, MAP, or MLLR adaptation are not effective at all.

Keywords

Kernel eigenvoice speaker adaptation, eigenvoice speaker adaptation, kernel principal component analysis, composite kernels, generalized EM algorithm

EDICS Category: 1-RECO

Corresponding Author: Dr. Brian Kan-Wing Mak.

The authors are with the Department of Computer Science, the Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong. E-mail: {mak,jamesk,csho}@cs.ust.hk

I. Introduction

In recent years, there has been a lot of interest in the study of kernel methods [1], [2], [3]. The basic idea is to map data in the input space $\mathcal{X}$ to a feature space[1] $\mathcal{F}$ via some nonlinear map φ, and then apply a linear method there. It is now well known that the computational procedure depends only on the inner products[2] $\varphi(\mathbf{x}_i)'\varphi(\mathbf{x}_j)$ in the feature space (where $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$), which can be obtained efficiently from a suitable kernel function $k(\cdot,\cdot)$. Thus, the use of kernels provides elegant nonlinear generalizations of many existing linear algorithms. A well-known example in supervised learning is the support vector machines (SVMs). In unsupervised learning, the kernel idea has also led to methods such as kernel-based clustering algorithms [4], kernel independent component analysis [5], and kernel principal component analysis [6]. In this paper, we would like to apply kernel methods to improve the performance of eigenvoice-based methods for fast speaker adaptation.

It is commonly known that a well-trained speaker-dependent (SD) model generally achieves a significantly lower word error rate than a speaker-independent (SI) model on recognizing speech from the specific speaker. For many applications such as phone services, it is hard to acquire a large amount of data from a user to train his/her SD model. A common technique to approach the SD performance is to adapt the SI model with a relatively small amount of SD speech using speaker adaptation methods. Adaptation methods like the Bayesian-based *maximum a posteriori* (MAP) adaptation [7] and the transformation-based *maximum likelihood linear regression* (MLLR) adaptation [8] have been popular for many years. Nevertheless, when the amount of available adaptation speech is really small — for example, only a few seconds, the more recent eigenvoice-based adaptation method is found particularly more effective. The (original) eigenvoice (EV) adaptation method [9] was motivated by the eigenface approach in face recognition [10]. The idea is to derive from a diverse set of speakers a small set of basis vectors called *eigenvoices* that are believed to represent different voice characteristics (e.g. gender, age, accent, etc.), and any training/new speaker is then a point in the eigenspace. In practice, a few to a few tens of eigenvoices are found adequate for fast speaker adaptation. Since the number of estimation parameters is greatly reduced, fast adaptation using EV is

[1] In the kernel methods terminology, the original space where raw data reside is called the *input space* and the space to which raw data are mapped is called the *feature space*. In order not to confuse this with the acoustic feature space in speech, the latter will always be called the "acoustic feature space", while the feature space in kernel methods will be simply called the "feature space" but may be sometimes called the "*kernel-induced feature space*" if additional clarity is necessary.

[2] In this paper, vector or matrix transpose is denoted by the superscript $'$.

possible with a few seconds of speech. The simple algorithm was later extended to work for large-vocabulary continuous speech recognition [11], [12], eigenspace-based MLLR [13], [14], and to approximate the model prior in MAP adaptation [15], [16], [17].

At the heart of eigenvoice-based adaptation methods is the principal component analysis (PCA) employed to find the eigenvoices. Then a new speaker is represented as a linear combination of a few most important eigenvoices, and the eigenvoice weights are usually estimated by maximizing the likelihood of the adaptation data. Traditionally, these eigenvoices are found by linear PCA. In this paper, we would like to exploit possible nonlinearity in the speaker supervector space, and investigate the use of nonlinear PCA to find the eigenvoices by kernel methods [6]. In effect, the nonlinear PCA problem is converted to a linear PCA problem in the high-dimensional kernel-induced feature space using the *kernel trick*. The eigenvoices thus derived will be called *kernel eigenvoices* (KEV), and we will call our new method *kernel eigenvoice speaker adaptation*. In principle, since the KEV adaptation method is a nonlinear generalization of the EV adaptation method, the former should be more powerful than the latter, and KEV adaptation is expected to give better performance. In fact, KEV adaptation will be reduced to the traditional EV adaptation method if linear kernel is employed.

One of the major challenges in KEV adaptation is to compute the state observation likelihoods of the speaker-adapted (SA) HMMs during the estimation of the kernel eigenvoice weights and subsequent decoding of the test speech. The reason is that unlike the conventional EV approach, the SA model found by KEV adaptation does *not* reside in the input speaker supervector space but in the kernel-induced feature space. Thus, in general, one cannot break up the SA model found by KEV adaptation into its constituent HMM Gaussians as in the EV approach. Our solution is the use of composite kernels in such a way that state observation likelihoods can be computed using only kernel functions without the need of an SA model in the input supervector space. Two different composite kernels, namely, direct sum kernel and tensor product kernel, are investigated. In addition, we also compare the performance of our novel KEV adaptation with that of EV, MAP, and MLLR adaptation methods.

Kernel eigenvoice will have to deal with several parameter spaces. To avoid confusion, we denote the several spaces as follows:

$\mathcal{O}$: d_1-dimensional observation space,

$\mathcal{X}$: d_2-dimensional input speaker supervector space,

$\mathcal{F}$: d_3-dimensional kernel-induced feature space.

In general, $d_1 \ll d_2 \ll d_3$. We will further put a "$\tilde{}$" on any quantity that has been centered in its respective space.

The rest of this paper is organized as follows. We first review the eigenvoice speaker adaptation method in Section II, and kernel principal component analysis in Section III. Then we will describe our new KEV adaptation method in details in Section IV, and its robust extension in Section V. In Section VI, we present the results of experimental evaluation of the KEV adaptation method using 2.1s, 4.1s, and 9.6s of adaptation speech, and compare it with EV, MAP, and MLLR adaptation methods. Finally, Section VII gives concluding remarks and some suggestions for future work.

II. Eigenvoice

In the standard eigenvoice speaker adaptation approach [9], speech training data are collected from many speakers with diverse characteristics. A set of *speaker-dependent* (SD) acoustic hidden Markov models (HMMs) are trained from each speaker where each HMM state is modeled as a mixture of Gaussian distributions. A speaker's voice is then represented by a *speaker supervector* that is composed by concatenating the mean vectors of all his/her HMM Gaussian distributions. For simplicity, we assume that each HMM state consists of one Gaussian only; the extension to mixtures of Gaussians is straightforward. Thus, the ith speaker supervector consists of R constituents, one from each Gaussian, and will be denoted by $\mathbf{x}_i = [\mathbf{x}'_{i1}, \ldots, \mathbf{x}'_{iR}]' \in \mathbb{R}^{d_2}$, where $d_2 = Rd_1$. The similarity between any two speaker supervectors $\mathbf{x}_i$ and $\mathbf{x}_j$ is measured by their inner product

$$\mathbf{x}'_i \mathbf{x}_j = \sum_{r=1}^{R} \mathbf{x}'_{ir} \mathbf{x}_{jr} \ . \tag{1}$$

PCA is then performed on a set of training speaker supervectors and the resulting eigenvectors are called *eigenvoices*. To adapt to a new speaker, his/her supervector $\mathbf{s}$ is treated as a linear combination of the first M eigenvoices $\{\mathbf{v}_1, \ldots, \mathbf{v}_M\}$ having the largest eigenvalues. That is, the centered supervector of the new speaker $\tilde{\mathbf{s}}$ is given by

$$\tilde{\mathbf{s}} \simeq \tilde{\mathbf{s}}^{(ev)} = \sum_{m=1}^{M} w_m \mathbf{v}_m \ , \tag{2}$$

where $\mathbf{w} = [w_1, \ldots, w_M]'$ is the eigenvoice weight vector. Usually, only a few eigenvoices (e.g., $M < 50$) are employed so that a little amount of adaptation speech (e.g., a few seconds) is required. Given the adaptation data $\boldsymbol{O} = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T\}$, the eigenvoice weights are usually estimated by maximizing the likelihood of $\boldsymbol{O}$. Mathematically, one finds $\mathbf{w}$ by *maximizing* the following $Q_{\pi ab}(\mathbf{w})$ function:

$$Q_{\pi ab}(\mathbf{w}) = Q_\pi + Q_a + Q_b(\mathbf{w}) \ , \tag{3}$$

where

$$\begin{aligned} Q_\pi &= \sum_{r=1}^{R} \gamma_1(r) \log(\pi_r) \ , & (4) \\ Q_a &= \sum_{p,r=1}^{R} \sum_{t=1}^{T-1} \xi_t(p,r) \log(a_{pr}) \ , & (5) \\ Q_b(\mathbf{w}) &= \sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_t(r) \log(b_r(\mathbf{o}_t, \mathbf{w})) \ , & (6) \end{aligned}$$

and π_r is the initial probability of state r; $\gamma_t(r)$ is the posterior probability of the observation sequence being at state r at time t; $\xi_t(p,r)$ is the posterior probability of the observation sequence being at state p at time t and at state r at time $t+1$; b_r is the Gaussian pdf of the rth state after re-estimation. Furthermore, $Q_b(\mathbf{w})$ is related to the new speaker supervector $\mathbf{s}$ by

$$Q_b(\mathbf{w}) = -\frac{1}{2} \sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_t(r) \left[d_1 \log(2\pi) + \log|\mathbf{C}_r| + \|\mathbf{o}_t - \mathbf{s}_r(\mathbf{w})\|^2_{\mathbf{C}_r} \right] , \tag{7}$$

where $\|\mathbf{o}_t - \mathbf{s}_r(\mathbf{w})\|^2_{\mathbf{C}_r} = (\mathbf{o}_t - \mathbf{s}_r(\mathbf{w}))' \mathbf{C}_r^{-1} (\mathbf{o}_t - \mathbf{s}_r(\mathbf{w}))$ and $\mathbf{C}_r$ is the covariance matrix of the Gaussian at state r. Since only the last term of Eqn. (7) depends on the eigenvoice weight vector $\mathbf{w}$, one may simply maximize the following reduced $Q(\mathbf{w})$ function to find the optimal $\mathbf{w}$:

$$Q(\mathbf{w}) = -\frac{1}{2} \sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_t(r) \|\mathbf{o}_t - \mathbf{s}_r(\mathbf{w})\|^2_{\mathbf{C}_r} \ . \tag{8}$$

By differentiating the reduced $Q(\mathbf{w})$ function with respect to $\mathbf{w}$, the optimal $\mathbf{w}$ can be found by solving a system of linear equations. Details can be found in [9].

III. Kernel Principal Component Analysis

In this paper, the computation of eigenvoices is generalized by performing kernel principal component analysis (PCA) instead of linear PCA. Linear PCA, on the other hand, can be considered as a special case of kernel PCA with the use of linear kernels. Fig. 1 gives an illustration of kernel PCA. Let $k(\cdot,\cdot)$ be the kernel with an associated mapping φ which maps a pattern $\mathbf{x} \in \mathbb{R}^{d_2}$ (a speaker supervector in the eigenvoice approach) in the input space $\mathcal{X}$ to $\varphi(\mathbf{x}) \in \mathbb{R}^{d_3}$ in the high-dimensional kernel-induced feature space $\mathcal{F}$. Given a set of N

patterns $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \in \mathcal{X}$, their φ-mapped feature vectors are $\{\varphi(\mathbf{x}_1), \ldots, \varphi(\mathbf{x}_N)\} \in \mathcal{F}$. The N mapped patterns are first centered in the feature space by finding the mean of the feature vectors $\bar{\varphi} = \frac{1}{N}\sum_{i=1}^{N} \varphi(\mathbf{x}_i)$. Let the "centered" map be $\tilde{\varphi}$ so that $\tilde{\varphi}(\mathbf{x}) = \varphi(\mathbf{x}) - \bar{\varphi}$. In addition, let $\mathbf{K} = [K_{ij}]$ be the kernel matrix with

$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)'\varphi(\mathbf{x}_j) \ , \tag{9}$$

and $\tilde{\mathbf{K}}$ be the centered version of $\mathbf{K}$ with $\tilde{K}_{ij} = \tilde{\varphi}(\mathbf{x}_i)'\tilde{\varphi}(\mathbf{x}_j)$. Notice that $\tilde{\mathbf{K}}$ is related to $\mathbf{K}$ by $\tilde{\mathbf{K}} = \mathbf{HKH}$, where $\mathbf{H} = \mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}'$ is the centering matrix, $\mathbf{I}$ is the $N \times N$ identity matrix, and $\mathbf{1} = [1, \ldots, 1]'$ is an N-dimensional vector.

To perform kernel PCA, instead of directly working on the covariance matrix in the feature space, one may carry out eigendecomposition on the centered kernel matrix $\tilde{\mathbf{K}}$ as

$$\tilde{\mathbf{K}} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}' \ , \tag{10}$$

where $\mathbf{U} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N]$ with $\boldsymbol{\alpha}_i = [\alpha_{i1}, \ldots, \alpha_{iN}]'$, and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$. The mth orthonormal eigenvector of the covariance matrix in the feature space is then given by ([6])

$$\mathbf{v}_m = \sum_{i=1}^{N} \frac{\alpha_{mi}}{\sqrt{\lambda_m}} \tilde{\varphi}(\mathbf{x}_i) \ . \tag{11}$$

Notice that all eigenvectors with non-zero eigenvalues are in the span of the φ-mapped data in the feature space.

IV. Kernel Eigenvoice (KEV)

As seen from Eqn. (8), an estimation of the eigenvoice weights requires the evaluation of the distances between adaptation data $\mathbf{o}_t$ and Gaussian means of the new speaker in the observation space $\mathcal{O}$. In the standard eigenvoice method, this is done by first breaking down the speaker-adapted supervector $\mathbf{s}^{(ev)}$ to obtain its R constituent Gaussian means $\mathbf{s}_1^{(ev)}, \ldots, \mathbf{s}_R^{(ev)}$. However, the use of kernel PCA does not allow us to access each constituent Gaussian directly. The reason is that in the standard EV approach, the state information is preserved during the concatenation of Gaussian mean vectors to form speaker supervectors; however, that piece of state information generally is lost during the φ-mapping of supervectors in the input space $\mathcal{X}$ to the high-dimensional feature space $\mathcal{F}$. Thus, in general, one cannot break up the speaker-adapted model found by KEV adaptation into its constituent HMM Gaussians as in the EV approach. To get around the problem, we investigate the use of composite kernels to preserve the necessary state information.

A. Definition of the Composite Kernel

For the ith speaker supervector $\mathbf{x}_i = [\mathbf{x}'_{i1}, \ldots, \mathbf{x}'_{iR}]'$, we map each constituent $\mathbf{x}_{ir}$ via a separate kernel $k_r(\cdot,\cdot)$ to $\varphi_r(\mathbf{x}_{ir})$, and construct $\varphi(\mathbf{x}_i)$ as $\varphi(\mathbf{x}_i) = [\varphi_1(\mathbf{x}_{i1})', \ldots, \varphi_R(\mathbf{x}_{iR})']'$. The similarity between two speaker supervectors $\mathbf{x}_i$ and $\mathbf{x}_j$ in the composite kernel-induced feature space $\mathcal{F}$ is measured by

$$k(\mathbf{x}_i, \mathbf{x}_j) = G(k_1(\mathbf{x}_{i1}, \mathbf{x}_{j1}), \ldots, k_R(\mathbf{x}_{iR}, \mathbf{x}_{jR})) \ , \tag{12}$$

where G is some function that combines the constituent kernels $k_r(\cdot,\cdot)$, $r = 1, \ldots, R$ into a valid composite kernel $k(\cdot,\cdot)$. Using this composite kernel, we can then proceed with the usual kernel PCA on the set of N training speaker supervectors and obtain the set of eigenvoices in the feature space $\mathcal{F}$ as given by Eqn. (11) in Section III.

A.1 Two Different Composite Kernels

In this paper, two different forms of composite kernel are investigated.

1. Direct sum kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sum_{r=1}^{R} k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr}) \ . \tag{13}$$

This may be the most intuitive form of composite kernels since

$$\begin{aligned} k(\mathbf{x}_i, \mathbf{x}_j) &= \varphi(\mathbf{x}_i)'\varphi(\mathbf{x}_j) \\ &= \begin{bmatrix} \varphi_1(\mathbf{x}_{i1}) \\ \vdots \\ \varphi_R(\mathbf{x}_{iR}) \end{bmatrix}' \begin{bmatrix} \varphi_1(\mathbf{x}_{j1}) \\ \vdots \\ \varphi_R(\mathbf{x}_{jR}) \end{bmatrix} \\ &= \sum_{r=1}^{R} \varphi_r(\mathbf{x}_{ir})'\varphi_r(\mathbf{x}_{jr}) \\ &= \sum_{r=1}^{R} k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr}) \ . \end{aligned}$$

2. Tensor product kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \prod_{r=1}^{R} k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr}) \ . \tag{14}$$

Furthermore, if the constituent kernels are Gaussian kernels

$$k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr}) = \exp(-\beta \|\mathbf{x}_{ir} - \mathbf{x}_{jr}\|^2_{\mathbf{C}_r}) \ , \tag{15}$$

where β is a tunable parameter that controls the width of the Gaussian kernels, then

$$\begin{aligned}
k(\mathbf{x}_i, \mathbf{x}_j) &= \prod_{r=1}^{R} k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr}) \\
&= \prod_{r=1}^{R} \exp(-\beta \|\mathbf{x}_{ir} - \mathbf{x}_{jr}\|^2_{\mathbf{C}_r}) \\
&= \exp\left(-\beta \sum_{r=1}^{R} \|\mathbf{x}_{ir} - \mathbf{x}_{jr}\|^2_{\mathbf{C}_r}\right) \\
&= \exp(-\beta \|\mathbf{x}_i - \mathbf{x}_j\|^2_{\mathbf{C}}) \ ,
\end{aligned}$$

where

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \vdots & \mathbf{0} & \mathbf{C}_R \end{bmatrix} .$$

That is, the tensor product kernel is then equivalent to a single Gaussian kernel with a block-diagonal covariance composed of the covariances from all constituent Gaussian kernels, $k_r(\cdot,\cdot),\ r = 1, \ldots, R$.

In both cases, if $k_r(\cdot,\cdot)$'s are valid kernels[3], so is $k(\cdot,\cdot)$ [2].

B. New Speaker in the Feature Space

Let the centered supervector of a new speaker found by kernel eigenvoice method in the feature space be $\tilde{\varphi}^{(kev)}(\mathbf{s})$. Conceptually, it corresponds to a speaker $\mathbf{s}$ in the input supervector space, even though $\mathbf{s}$ may not exist[4]. However, our KEV adaptation method does not require the existence of the pre-image $\mathbf{s}$ in the input supervector space.

Analogous to the formulation of a new speaker in the standard eigenvoice approach (Eqn. (2)), $\tilde{\varphi}^{(kev)}(\mathbf{s})$ is assumed to be a linear combination of the first M eigenvectors with the largest eigenvalues found by kernel PCA in $\mathcal{F}$. That is,

$$\tilde{\varphi}^{(kev)}(\mathbf{s}) = \sum_{m=1}^{M} w_m \mathbf{v}_m = \sum_{m=1}^{M} \sum_{i=1}^{N} \frac{w_m \alpha_{mi}}{\sqrt{\lambda_m}} \tilde{\varphi}(\mathbf{x}_i) \ . \tag{16}$$

[3] Valid kernel functions are those that satisfy Mercer's theorem [18]. It then follows that there exist a feature space $\mathcal{F}$ and a mapping φ corresponding to the kernel k such that scalar products of the form $\varphi(\mathbf{x}_i)'\varphi(\mathbf{x}_j)$ in the feature space $\mathcal{F}$ can be computed as $k(\mathbf{x}_i, \mathbf{x}_j)$.

[4] The notation for a new speaker in the feature space requires some explanation. If $\mathbf{s}$ exists, then its centered image is $\tilde{\varphi}^{(kev)}(\mathbf{s})$. However, since the pre-image of a speaker found in the feature space may not exist [2], the notation $\tilde{\varphi}^{(kev)}(\mathbf{s})$ is not exactly correct. However, the notation is adopted for its intuitiveness and the readers are advised to infer the existence of $\mathbf{s}$ based on the context.

Its rth constituent is then given by

$$\tilde{\varphi}_r^{(kev)}(\mathbf{s}_r) = \sum_{m=1}^{M}\sum_{i=1}^{N} \frac{w_m \alpha_{mi}}{\sqrt{\lambda_m}} \tilde{\varphi}_r(\mathbf{x}_{ir}) \ .$$

Hence, the similarity between $\varphi_r^{(kev)}(\mathbf{s}_r)$ and $\varphi_r(\mathbf{o}_t)$ is given by

$$\begin{aligned}
k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) &\equiv \varphi_r^{(kev)}(\mathbf{s}_r)' \varphi_r(\mathbf{o}_t) \\
&= \left[\left(\sum_{m=1}^{M}\sum_{i=1}^{N} \frac{w_m \alpha_{mi}}{\sqrt{\lambda_m}} \tilde{\varphi}_r(\mathbf{x}_{ir}) \right) + \bar{\varphi}_r \right]' \varphi_r(\mathbf{o}_t) \\
&= \left[\left(\sum_{m=1}^{M}\sum_{i=1}^{N} \frac{w_m \alpha_{mi}}{\sqrt{\lambda_m}} (\varphi_r(\mathbf{x}_{ir}) - \bar{\varphi}_r) \right) + \bar{\varphi}_r \right]' \varphi_r(\mathbf{o}_t) \\
&= \sum_{m=1}^{M}\sum_{i=1}^{N} \frac{w_m \alpha_{mi}}{\sqrt{\lambda_m}} \left(k_r(\mathbf{x}_{ir}, \mathbf{o}_t) - \bar{\varphi}_r' \varphi_r(\mathbf{o}_t) \right) + \bar{\varphi}_r' \varphi_r(\mathbf{o}_t) \\
&\equiv A(r,t) + \sum_{m=1}^{M} \frac{w_m}{\sqrt{\lambda_m}} B(m,r,t) \ , \qquad (17)
\end{aligned}$$

where $\bar{\varphi}_r = \frac{1}{N}\sum_{i=1}^{N} \varphi_r(\mathbf{x}_{ir})$ is the rth part of $\bar{\varphi}$,

$$A(r,t) = \bar{\varphi}_r' \varphi_r(\mathbf{o}_t) = \frac{1}{N} \sum_{j=1}^{N} k_r(\mathbf{x}_{jr}, \mathbf{o}_t) \ , \qquad (18)$$

and

$$B(m,r,t) = \left(\sum_{i=1}^{N} \alpha_{mi} k_r(\mathbf{x}_{ir}, \mathbf{o}_t) \right) - A(r,t) \left(\sum_{i=1}^{N} \alpha_{mi} \right) . \qquad (19)$$

Furthermore, the derivative of $k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t)$ with respect to each eigenvoice weight w_j, $j = 1, \ldots, M$, is given by

$$\frac{\partial}{\partial w_j} \left(k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) \right) = \frac{B(j,r,t)}{\sqrt{\lambda_j}} \ , \qquad (20)$$

which will be needed for the maximum likelihood estimation of the eigenvoice weights.

C. Maximum Likelihood Adaptation Using an Isotropic Kernel

On adaptation, we have to express $\|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r}$ of Eqn. (8) as a function of $\mathbf{w}$. Consider using isotropic kernels for k_r so that $k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr}) = \kappa(\|\mathbf{x}_{ir} - \mathbf{x}_{jr}\|^2_{\mathbf{C}_r})$. Then $k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) = \kappa(\|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r})$, and if κ is invertible, $\|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r}$ will be a function of $k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t)$, which in turn is a function of $\mathbf{w}$ by Eqn. (17). In the sequel, we will use the Gaussian kernel of Eqn. (15), and hence

$$\begin{aligned}
k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) &= \exp(-\beta \|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r}) \\
\Rightarrow \quad \|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r} &= -\frac{1}{\beta} \log \left(k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) \right) \ . \qquad (21)
\end{aligned}$$

Substituting Eqn. (21) to the reduced $Q(\mathbf{w})$ function of Eqn. (8), and differentiating $Q(\mathbf{w})$ with respect to each eigenvoice weight, $w_j,\ j = 1, \ldots, M$, we get

$$\frac{\partial Q}{\partial w_j} = \frac{1}{2\beta} \sum_{r=1}^{R} \sum_{t=1}^{T} \frac{\gamma_t(r)}{k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t)} \cdot \frac{\partial}{\partial w_j} \left(k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) \right) \ . \tag{22}$$

By making use of the gradient in Eqn. (20), we obtain

$$\frac{\partial Q}{\partial w_j} = \frac{1}{2\beta\sqrt{\lambda_j}} \sum_{r=1}^{R} \sum_{t=1}^{T} \frac{\gamma_t(r) B(j, r, t)}{k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t)} \ . \tag{23}$$

D. Generalized EM Algorithm

Because of the nonlinear nature of kernel PCA, Eqn. (23) is nonlinear in $\mathbf{w}$ and there is no closed form solution for the optimal $\mathbf{w}$. In this paper, we apply the generalized EM algorithm (GEM) [19] to find the optimal weights. GEM is similar to the conventional EM except for the maximization step: EM looks for a $\mathbf{w}$ that maximizes the expected likelihood of the E-step but GEM only requires a $\mathbf{w}$ that improves the likelihood. Many numerical methods [20] may be used to update $\mathbf{w}$ based on the derivatives of Q. In this paper, gradient ascent is used to get $\mathbf{w}(n)$ from $\mathbf{w}(n-1)$ based only on the first-order derivative: $\mathbf{w}(n) = \mathbf{w}(n-1) + \eta(n)\mathbf{Q}'|_{\mathbf{w}=\mathbf{w}(n-1)}$, where $\mathbf{Q}' = \frac{\partial Q}{\partial \mathbf{w}}$ and $\eta(n)$ is the learning rate at the nth iteration. Methods such as the Newton's method that uses the second-order derivatives may also be used for faster convergence at the expense of computing the more costly Hessian in each iteration.

The initial value of $\mathbf{w}(0)$ can be important for numerical methods like gradient ascent. One reasonable approach is to start with the eigenvoice weights of the supervector composed from the speaker-independent model $\mathbf{x}^{(si)}$. That is, for $m = 1, \ldots, M$,

$$\begin{aligned} w_m(0) &= \mathbf{v}_m' \tilde{\varphi}(\mathbf{x}^{(si)}) \\ &= \sum_{i=1}^{N} \frac{\alpha_{mi}}{\sqrt{\lambda_m}} \tilde{\varphi}(\mathbf{x}_i)' \tilde{\varphi}(\mathbf{x}^{(si)}) \\ &= \sum_{i=1}^{N} \frac{\alpha_{mi}}{\sqrt{\lambda_m}} (\varphi(\mathbf{x}_i) - \bar{\varphi})'(\varphi(\mathbf{x}^{(si)}) - \bar{\varphi}) \\ &= \sum_{i=1}^{N} \frac{\alpha_{mi}}{\sqrt{\lambda_m}} \left[k(\mathbf{x}_i, \mathbf{x}^{(si)}) + \frac{1}{N^2} \sum_{p,q=1}^{N} k(\mathbf{x}_p, \mathbf{x}_q) - \frac{1}{N} \sum_{p=1}^{N} \big(k(\mathbf{x}_i, \mathbf{x}_p) + k(\mathbf{x}^{(si)}, \mathbf{x}_p) \big) \right] \ . \end{aligned} \tag{24}$$

V. Robust Kernel Eigenvoice

The success of the eigenvoice approach for fast speaker adaptation is due to two factors: (1) a good collection of "diverse" speakers so that the whole speaker space is captured by the eigenvoices; and (2) the number of adaptation parameters is reduced to a few eigenvoice weights. However, since the amount of adaptation data is so small, the adaptation performance may vary widely. To get a more robust performance, we propose to interpolate the kernel eigenvoice $\tilde{\varphi}^{(kev)}(\mathbf{s})$ with the $\tilde{\varphi}$-mapped speaker-independent (SI) supervector $\tilde{\varphi}(\mathbf{x}^{(si)})$ to obtain the final speaker-adapted model (in the feature space) $\tilde{\varphi}^{(rkev)}(\mathbf{s})$ as follows:

$$\tilde{\varphi}^{(rkev)}(\mathbf{s}) = w_0\tilde{\varphi}(\mathbf{x}^{(si)}) + (1 - w_0)\tilde{\varphi}^{(kev)}(\mathbf{s}),\ 0.0 \le w_0 \le 1.0 \ , \tag{25}$$

where $\tilde{\varphi}^{(kev)}(\mathbf{s})$ is found by Eqn. (16). Following similar mathematical treatment as in Section IV-B, it can be shown that the similarity between the $\varphi_r^{(rkev)}(\mathbf{s}_r)$ and $\varphi_r(\mathbf{o}_t)$ is given by

$$\begin{aligned} k_r^{(rkev)}(\mathbf{s}_r, \mathbf{o}_t) &\equiv \varphi_r^{(rkev)}(\mathbf{s}_r)' \varphi_r(\mathbf{o}_t) \\ &= w_0 k_r(\mathbf{x}_r^{(si)}, \mathbf{o}_t) + (1 - w_0) k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) \ . \end{aligned} \tag{26}$$

Hence, the gradients required to estimate w_0 jointly with other eigenvoice weights are

$$\frac{\partial}{\partial w_0}\left(k_r^{(rkev)}(\mathbf{s}_r, \mathbf{o}_t)\right) = k_r(\mathbf{x}_r^{(si)}, \mathbf{o}_t) - k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t) \tag{27}$$

and

$$\frac{\partial}{\partial w_j}\left(k_r^{(rkev)}(\mathbf{s}_r, \mathbf{o}_t)\right) = (1 - w_0)\frac{\partial}{\partial w_j}\left(k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t)\right) \ , \ j = 1, \ldots, M \ . \tag{28}$$

The two gradients can be evaluated using the results in Eqn. (17) and Eqn. (20) of Section IV-B respectively.

Notice that $\tilde{\varphi}^{(rkev)}(\mathbf{s})$ also contains components in $\tilde{\varphi}(\mathbf{x}^{(si)})$ from eigenvectors beyond the M selected kernel eigenvoices for adaptation. Thus, robust KEV adaptation may have the additional benefit of preserving the speaker-independent projections on the remaining less important but possibly robust eigenvoices in the final speaker-adapted model.

VI. Experimental Evaluation

The proposed kernel eigenvoice adaptation method was evaluated on the TIDIGITS speech corpus [21]. We first studied the number of kernel eigenvoices for best performance in this recognition task, and the effectiveness of the two forms of composite kernels. Then the performance of our new kernel eigenvoice adaptation was compared with that of the speaker-independent models, the conventional eigenvoice adaptation, MAP adaptation, and MLLR adaptation.

A. TIDIGITS Corpus

The TIDIGITS corpus contains clean connected-digit utterances sampled at 20 kHz. It is divided into a standard training set and a test set. There are 163 speakers (of both genders) in each set, each pronouncing 77 utterances of one to seven digits (out of the eleven digits: "0", "1", . . ., "9", and "oh"). There is no overlap between the training speakers and test speakers. The speaker characteristics are quite diverse with speakers coming from 22 dialect regions of USA, and their ages ranging from 6 to 70 years old.

B. Acoustic Models

All training data were processed to extract 12 mel-frequency cepstral coefficients and the normalized frame energy from each speech frame of 25 ms at every 10 ms. Each of the eleven digit models was a strictly left-to-right HMM comprising 16 states and one Gaussian with diagonal covariance per state. In addition, there were a 3-state "sil" model to capture silence speech and a 1-state "sp" model to capture short pauses between digits. All HMMs were trained by the EM algorithm. Thus, the dimension of the observation space d_1 is 13 and that of the speaker supervector space d_2 is 11 models × 16 states/model × 13/state = 2288.

Firstly, a set of speaker-independent (SI) digit models were trained. Then a set of speaker-dependent (SD) digit models were trained for each individual training speaker by borrowing the variances and transition matrices from the corresponding SI models, and only the Gaussian means were estimated. Furthermore, the "sil" and "sp" models were simply copied to the SD model. In pilot experiments, it was found that SD models trained in this way performed better than SD models that did not share any model parameters with the SI models.

C. Experiments

In all experiments, only the training set was used to train the SI HMMs and SD HMMs from which the SI and SD speaker supervectors were derived. Adaptation was performed on the test speakers. Five, ten, and twenty digits were used for adaptation, which correspond to an average of 2.1s, 4.1s, and 9.6s of adaptation speech (or 3.0s, 5.5s, and 13.0s of speech if the leading and ending silences are counted). To improve the statistical reliability of the results, all results were the averages of 5-fold cross-validation over all 163 test speakers. Moreover, except for one experiment, all other adaptation experiments were performed in supervised mode.

The following models/systems are compared:

SI*:* the baseline speaker-independent model.

KEV*:* the speaker-adapted model found by our new kernel eigenvoice adaptation method as described in Section IV.

Robust-KEV*:* the speaker-adapted model found by our robust KEV adaptation method as described in Section V.

EV*:* the speaker-adapted model found by the standard eigenvoice adaptation method as described in [9].

Robust-EV*:* the speaker-adapted model computed as the interpolation between the SI supervector and the supervector found by EV adaptation. That is,

$$\mathbf{s}^{(rev)} = w_0\mathbf{x}^{(si)} + (1 - w_0)\mathbf{s}^{(ev)} \ , \quad 0.0 \leq w_0 \leq 1.0 \ , \tag{29}$$

where $\mathbf{s}^{(rev)}$ is the final speaker supervector found by robust EV adaptation, and w_0 is estimated jointly with the other eigenvoice weights by maximizing the likelihood of adaptation data. This is analogous to the robust KEV adaptation.

MAP*:* the speaker-adapted model found by MAP adaptation.

MLLR*:* the speaker-adapted model found by MLLR adaptation.

Before we describe our experiments on KEV adaptation, remarks on some experimentation issues are worth mentioning:

- There is one tunable parameter β in the composite Gaussian kernels. A suitable value of β was searched as follows: 10 speakers were randomly chosen from the training set for KEV adaptation; 4.1s of adaptation data were used, and the best value of β was empirically determined to be around 0.0005. This value of β was used in all reported experiments.
- The learning rate was initially set to 0.0001.
- The word accuracy of the baseline SI model on the test data is 96.25%[5].
- To check the quality of our SD models, a 7-fold cross-validation was performed: for each training speaker, his data was divided into 7 roughly equal subsets, and 6 subsets were used for training his acoustic model which was then tested on the remaining subset. The average word accuracy over all training speakers is found to be 98.76%. It shows that our way of training SD models produces sufficiently good acoustic models for subsequent eigenvoice determination.

[5] The word accuracy of our SI model is not as good as the best reported result on TIDIGITS which is about 99.7%. The main reasons are that we used only 13-dimensional static cepstra and energy, and each state was modeled by a single Gaussian with diagonal covariance. The use of this simple model allowed us to run experiments with 5-fold cross-validation using very short adaptation speech. Right now our approach requires online computation of many kernel function values and is very computationally expensive. As a first attempt on the approach, we feel that the use of this simple model is justified. We are now working on its speed-up (as is discussed in the future works in Section VII) and its extension to HMM states of Gaussian mixtures.

C.1 Experiment 1: Number of Kernel Eigenvoices

Fig. 2 shows the detailed results of (robust) KEV adaptation using various numbers of kernel eigenvoices. The direct sum composite kernel was employed, and only the results from using 2.1s and 9.6s of adaptation speech are shown in the figure. The results show that KEV adaptation can outperform the SI model even with only two eigenvoices using only 2.1s of speech. Its performance then improves slightly with more eigenvoices or more adaptation data. If we allow interpolation with the SI model as in robust KEV adaptation, the performance improvement as well as the saturation effect are even more pronounced: even with one eigenvoice, the adaptation performance is already better than that of the SI model, and then the performance does not change much with more eigenvoices or adaptation data. The results seem to suggest that the requirement that the adapted speaker supervector is a weighted sum of few eigenvoices is both the strength and weakness of the method: on the one hand, fast adaptation becomes possible since the number of estimation parameters is small, but adaptation saturates quickly because the constraint is so restrictive that all mean vectors of different acoustic models have to undergo the same linear combination of the eigenvoices.

Moreover, the interpolation with the SI model in robust KEV adaptation significantly improves the performance of KEV adaptation. The improvement is more pronounced with fewer adaptation data.

From Fig. 2, the optimal number of kernel eigenvoices for this task is 8, and this is the number of kernel eigenvoices used in the (robust) KEV adaptation experiments for the rest of this paper.

C.2 Experiment 2: Direct Sum Kernel vs. Tensor Product Kernel

The two types of composite kernels, namely, direct sum kernel and tensor product kernel, were compared using the robust KEV adaptation. The results are shown in Table I. It can be seen that there is no significant difference between their performance. Therefore, we simply pick one of them — the direct sum composite kernel — for the remaining (robust) KEV adaptation experiments.

C.3 Experiment 3: KEV vs. EV, MAP, MLLR

In this experiment, KEV adaptation was compared with several other adaptation methods, EV, MAP, and MLLR adaptation. For each adaptation method, efforts were made to find the best setup for the method so as to obtain its best results for comparison purpose. That means, for EV or KEV adaptation (using direct sum composite kernel), the best results were obtained with the optimal number of eigenvoices; for MAP adaptation, the best results were achieved with the optimal scaling factors; for MLLR adaptation, only global MLLR was tried, and the better results from using either block-diagonal or full transformation matrices were used for comparison. The results are plotted in Fig. 3.

If we only look at the (robust) EV and (robust) KEV performance, it is clear that our (robust) KEV adaptation always performs better than (robust) EV adaptation. The results show that nonlinear kernel PCA using composite kernels can be more effective in finding the eigenvoices. Moreover, the incorporation of the SI prior information in EV or KEV adaptation always improves the adaptation performance.

When all adaptation methods are compared, we observe that when only 2.1s or 4.1s of adaptation data are available, EV adaptation and MAP adaptation have similar performance; MLLR adaptation, SI models, and robust EV adaptation are better with similar performance. However, only our new KEV and robust KEV adaptation work significantly better than the SI model; EV, MAP, and MLLR adaptation all perform worse than the SI model, and robust EV adaptation can only match the SI performance in this task. Only for the case with 9.6s of adaptation data, then MLLR works marginally better than the robust KEV method by an absolute 0.06%.

Specifically, KEV adaptation obtains a word error rate (WER) reduction of 16.0%, 21.3%, and 21.3% with 2.1s, 4.1s, and 9.6s of adaptation speech over the SI model. When the SI model is interpolated with the KEV model in our robust KEV method, the WER reduction further increases to 27.5%, 31.7%, and 33.3% respectively.

Analysis of Eigenvoices

The conventional eigenvoice adaptation method does not seem to be effective in this task. A detailed examination reveals that the performance of EV adaptation does not change much with increasing number of eigenvoices. For instance, with 10s of adaptation data, the adaptation performance using 1 to 5 eigenvoices differs only by 0.2%. In fact, the optimal number of eigenvoices in EV adaptation is one. Kim *et al.* also had similar findings in their adaptation experiments using continuous Korean digits [15].

We further analyze the eigenvoices found the EV and KEV adaptation methods. Fig. 4

shows the distribution of all 163 test speakers in the subspace spanned by the first two leading eigenvoices found by EV. It can be seen that although Women and Girls can be fairly separated by the first eigenvoice, in general, there are significant overlaps among the four groups of speakers: Men, Women, Boys, and Girls. This may explain why one eigenvoice is adequate for this task. A similar plot is prepared for KEV adaptation, and the result is shown in Fig. 5. It is now noticed that there are clear separations among the three groups: Men, Women, and Girls, and the latter two groups overlap with the Boys. From the data representation point of view, the KEV adaptation method clearly produces better eigenvoices than the EV adaptation method to represent the four different groups of speakers.

C.4 Experiment 4: Supervised vs. Unsupervised Adaptation

In all the above experiments, adaptation was performed in supervised manner. In this last experiment, we would like to see the effect of unsupervised adaptation. Robust KEV adaptation using the direct sum composite kernel was run in both supervised and unsupervised modes on the same data, and the results are shown in Table II. We can see that the performance of unsupervised robust KEV adaptation is only slightly worse than that of its supervised counterpart. The results are expected since the SI model is already quite accurate.

VII. Conclusions and Future Works

In this paper, we study the use of kernel PCA with a composite kernel to derive better eigenvoices to improve the standard eigenvoice speaker adaptation. Two forms of composite kernels, namely, direct sum kernel and tensor product kernel, are investigated. In the TIDIGITS task, it is found that while the standard eigenvoice approach does not help, our kernel eigenvoice method may outperform the speaker-independent model by about 16–21% (in terms of word error rate reduction). Moreover, we also propose to interpolate the speaker-independent model with the speaker model found by our kernel eigenvoice approach in the robust kernel eigenvoice adaptation. The robust extension leads to 28–33% word error rate reduction over the performance of the SI model.

Although kernel PCA elegantly introduces nonlinearity in the linear PCA procedure, and renders kernel eigenvoice adaptation more powerful than the standard eigenvoice adaptation, there is a price to pay: online computation of many kernel functions is required during subsequent speech recognition. To understand this, one should notice that the computation of state observation likelihoods requires the evaluation of the distance $\|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r}$ of Eqn. (21). The distance now has to be computed via the kernel value of $k_r^{(kev)}(\mathbf{s}_r, \mathbf{o}_t)$ given by Eqn. (17). As a result, our new *kernel eigenvoice* adaptation method is slower than the standard eigenvoice adaptation method during both adaptation and recognition. We are pursuing two possible solutions:

- reducing the number of kernel functions to compute. One possible solution is to apply *sparse kernel PCA* [22] so that the computation of the first M principal components involves only M (instead of N with $M \ll N$) kernel functions. Another solution is to use *compactly supported kernels* [23], in which the value of $\kappa(\|\mathbf{x}_i - \mathbf{x}_j\|)$ vanishes when $\|\mathbf{x}_i - \mathbf{x}_j\|$ is greater than a certain threshold. The kernel matrix then becomes sparse. Moreover, no more computation is required when $\|\mathbf{x}_i - \mathbf{x}_j\|$ is large.
- eliminating the need to compute any kernel functions during recognition. This can be achieved if we can map the speaker-adapted model found by kernel eigenvoice adaptation in the feature space back to its *pre-image* speaker supervector in the input space. Although an exact pre-image generally does not exist, recently, we have developed a closed-form solution for finding an approximate pre-image [24] in an image denoising problem that uses kernel PCA. We will adopt the algorithm in [24] to find an approximate speaker supervector in the input space for our KEV adaptation method, and study its performance on test speech.

Acknowledgements

This research is partially supported by the Research Grants Council of the Hong Kong SAR under the grant numbers HKUST6195/02E, HKUST6201/02E, and CA02/03.EG04.

References

[1] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*, Cambridge University Press, 2000.

[2] B. Schölkopf and A.J. Smola, *Learning with Kernels*, MIT Press, Cambridge, MA, 2002.

[3] V. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.

[4] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik, “Support vector clustering,” *Journal of Machine Learning Research*, vol. 2, pp. 125–137, 2001.

[5] F. R. Bach and M. I. Jordan, “Kernel independent component analysis,” *Journal of Machine Learning Research*, vol. 3, pp. 1–48, 2002.

[6] B. Schölkopf, A. Smola, and K. R. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Computation*, vol. 10, pp. 1299–1319, 1998.

[7] J. L. Gauvain and C. H. Lee, “Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains,” *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, April 1994.

[8] C. J. Leggetter and P. C. Woodland, “Maximum likelihood linear regression for speaker adaptation of

continuous density hidden Markov models," *Journal of Computer Speech and Language*, vol. 9, pp. 171–185, 1995.

[9] R. Kuhn, J.-C. Junqua, P. Nguyen, and N. Niedzielski, "Rapid speaker adaptation in eigenvoice space," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 4, pp. 695–707, Nov 2000.

[10] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1991, pp. 586–591.

[11] R. Kuhn, F. Perronnin, P. Nguyen, J. C. Junqua, and L. Rigazio, "Very fast adaptation with a compact context-dependent eigenvoice model," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2001, vol. 1, pp. 373–376.

[12] H. Botterweck, "Very fast adaptation for large vocabulary continuous speech recognition using eigenvoices," in *Proceedings of the International Conference on Spoken Language Processing*, 2000, vol. 4, pp. 354–357.

[13] K. T. Chen, W. W. Liau, H. M. Wang, and L. S. Lee, "Fast speaker adaptation using eigenspace-based maximum likelihood linear regression," in *Proceedings of the International Conference on Spoken Language Processing*, 2000, vol. 3, pp. 742–745.

[14] N. Wang, S. Lee, F. Seide, and L. S. Lee, "Rapid speaker adaptation using a priori knowledge by eigenspace analysis of MLLR parameters," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2001, pp. 345–348.

[15] D. K. Kim and N. S. Kim, "Bayesian speaker adaptation based on probabilistic principal component analysis," in *Proceedings of the International Conference on Spoken Language Processing*, 2000, pp. 734–737.

[16] E. Jon, D. K. Kim, and N. S. Kim, "EMAP-based speaker adaptation with robust correlation estimation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2001, pp. 321–324.

[17] H. Botterweck, "Anisotropic MAP defined by eigenvoices for large vocabulary continuous speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2001, vol. 1, pp. 353–356.

[18] J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Transactions of the London Philosophical Society A*, vol. 209, pp. 415–446, 1909.

[19] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[20] G.J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, Wiley, New York, 1996.

[21] R. G. Leonard, "A database for speaker-independent digit recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1984, vol. 3, pp. 4211–4214.

[22] A. J. Smola, O. L. Mangasarian, and B. Schölkopf, "Sparse kernel feature analysis," Tech. Rep. 99-03, Data Mining Institute, University of Wisconsin, Madison, 1999.

[23] M. G. Genton, "Classes of kernels for machine learning: A statistics perspective," *Journal of Machine Learning Research*, vol. 2, pp. 299–312, 2001.

[24] J. T. Kwok and I. W. Tsang, "The pre-image problem in kernel methods," in *Proceedings of the 20th International Conference on Machine Learning*, Washington, D.C., USA, August 2003, pp. 408–415.

TABLE I

Performance of direct sum kernel and tensor product kernel in robust KEV adaptation. Results are word recognition accuracies.

Type of Composite Kernel	2.1s	4.1s	9.6s
direct sum kernel	97.28%	97.44%	97.50%
tensor product kernel	97.33%	97.42%	97.43%

TABLE II

Supervised vs. unsupervised robust KEV adaptation using direct sum kernel. Results are word recognition accuracies.

Adaptation Mode	2.1s	4.1s	9.6s
Supervised	97.28%	97.44%	97.50%
Unsupervised	97.22%	97.27%	97.34%

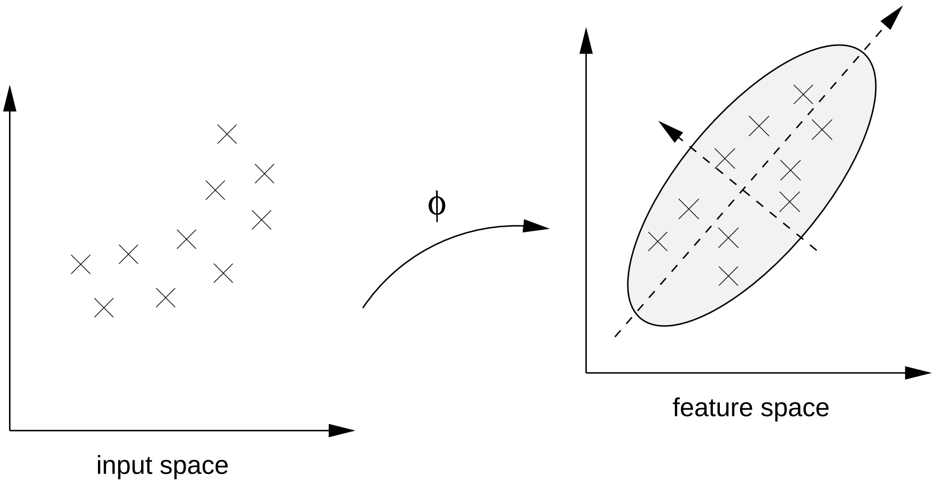

Fig. 1. Illustration of kernel PCA.

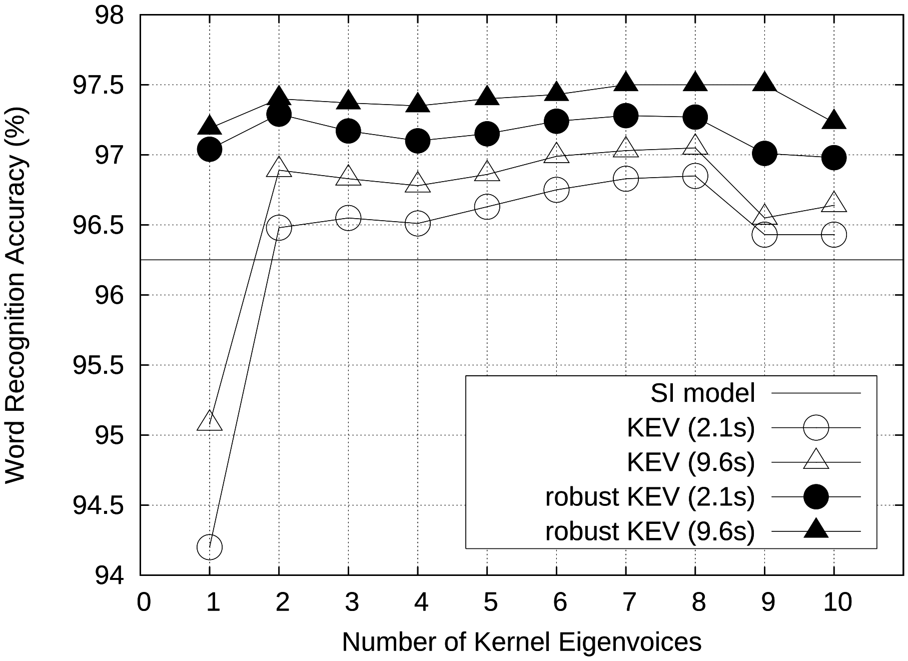

Fig. 2. Performance of KEV and robust KEV adaptation using direct sum composite kernel and different numbers of kernel eigenvoices.

e

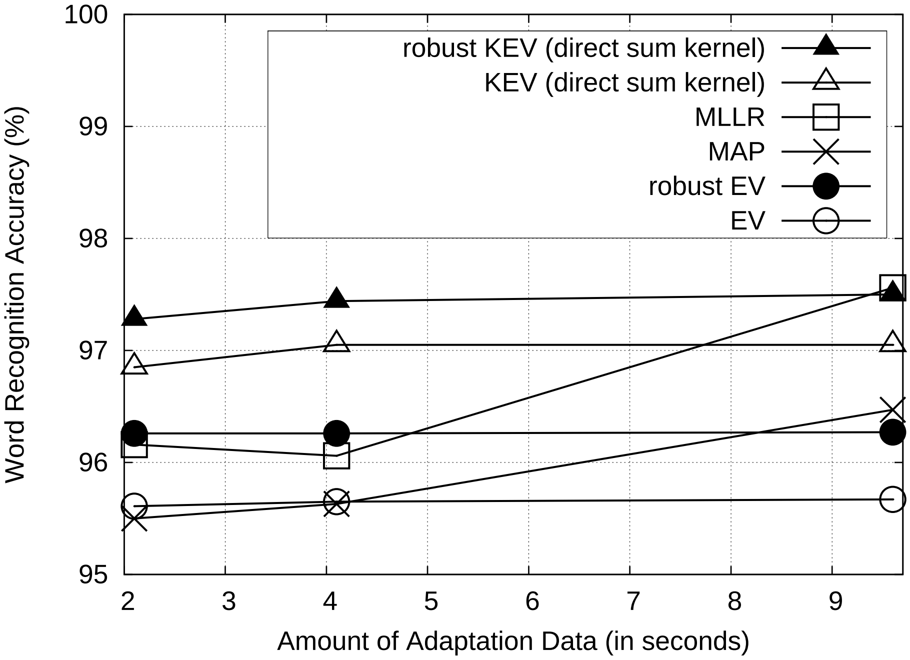

Fig. 3. Performance comparison among EV, KEV (with direct sum composite kernel), MAP, and MLLR adaptation methods. (Recall that the accuracy of the baseline SI model is 96.25%.)

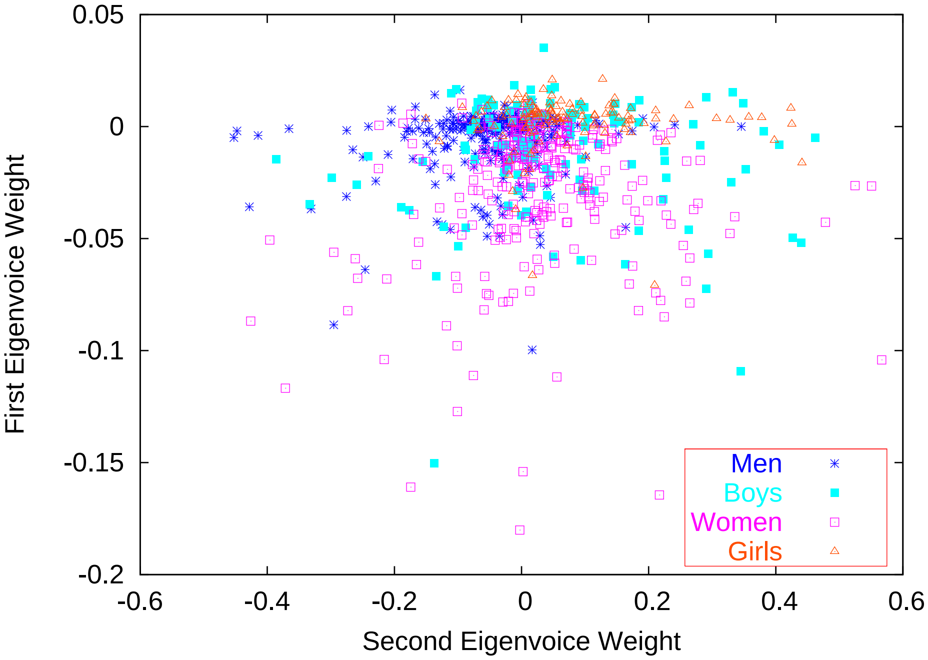

Fig. 4. Distribution of the 163 test speakers on the subspace spanned by the top two eigenvoices found by EV adaptation.

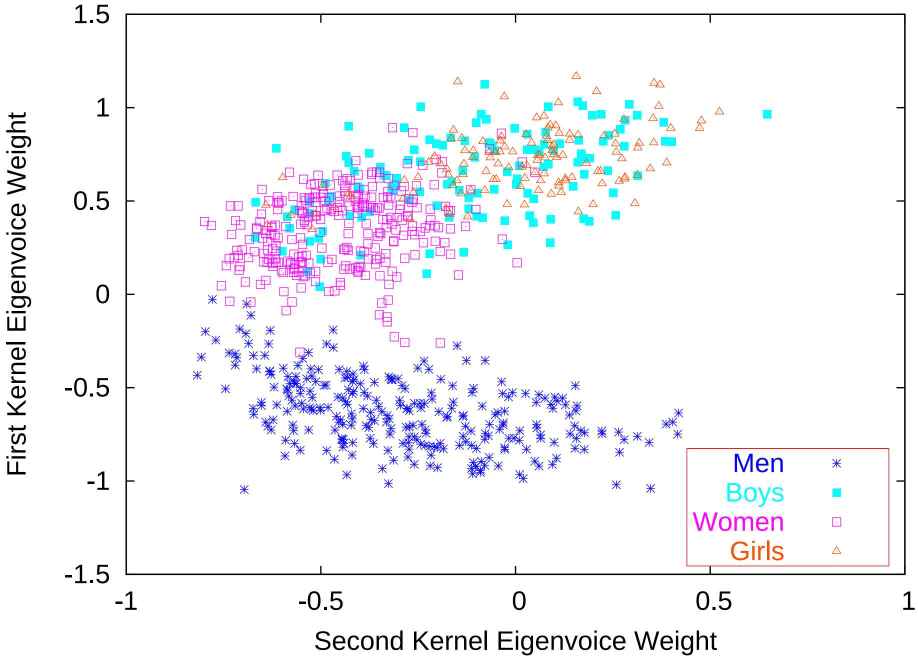

Fig. 5. Distribution of the 163 test speakers on the subspace spanned by the top two kernel eigenvoices found by KEV adaptation.

List of Tables

List of Figures