Statistical Parametric Speech Synthesis Based on Speaker and Language Factorization

Heiga Zen, *Member, IEEE*, Norbert Braunschweiler, Sabine Buchholz, Mark J. F. Gales, *Fellow, IEEE*, Kate Knill, *Member, IEEE*, Sacha Krstulović, and Javier Latorre, *Member, IEEE*

***Abstract*—An increasingly common scenario in building speech synthesis and recognition systems is training on inhomogeneous data. This article proposes a new framework for estimating hidden Markov models on data containing both multiple speakers and multiple languages. The proposed framework, *speaker and language factorization*, attempts to factorize speaker-/language-specific characteristics in the data and then model them using separate transforms. Language-specific factors in the data are represented by transforms based on cluster mean interpolation with cluster-dependent decision trees. Acoustic variations caused by speaker characteristics are handled by transforms based on constrained maximum likelihood linear regression. Experimental results on statistical parametric speech synthesis show that the proposed framework enables data from multiple speakers in different languages to be used to: train a synthesis system; synthesize speech in a language using speaker characteristics estimated in a different language; adapt to a new language.**

***Index Terms*—speaker and language factorization, hidden Markov models, statistical parametric speech synthesis**

I. Introduction

MANY different factors influence speech signals, including the words being uttered, the speaker, the language, and the speaking style. To handle variations caused by these factors in acoustic modeling for automatic speech recognition (ASR), the concept of *adaptive training* was introduced [2], [4]. Here a transform is associated with each homogeneous block in the data, such as a speaker in a particular noise condition. A canonical model set is trained given these transforms. This concept was then extended so that each of the factors affecting the speech signals is modelled separately, referred to as *acoustic factorization* [6]. Here a separate transform is generated for each factor then a canonical model set is built given the combined transforms for all factors. For example, speaker and noise factors have been considered [6], [27].

Recently, statistical parametric speech synthesis [40] based on hidden Markov models (HMMs) has grown in popularity in text-to-speech (TTS). In this approach, spectra, excitations, and durations of speech are modelled in a unified framework of context-dependent sub-word HMMs [34]. For a given text to be synthesized, speech parameter trajectories that maximize their output probabilities are generated from the trained HMMs [24]. This approach has various advantages over the concatenative speech synthesis approach, such as the flexibility to change its voice characteristics [17], [30]. To use speech data from multiple speakers to increase the amount of training data, adaptive training has been introduced to statistical parametric speech synthesis [30]. This article examines an application of acoustic factorization to statistical parametric speech synthesis, where speaker and language factors are considered.

The two primary factors that influence speech are the voice characteristics of the speaker and the language spoken. By representing the voice characteristics by one transform and language attributes by a completely separate transform, the synthesis system will be able to alter language, or speaker, separately. Thus factoring out speaker and language yields a number of options for synthesis. Firstly, it can be applied in polyglot speech synthesis. Unlike traditional multilingual speech synthesis systems, which share common algorithms for all languages [21], in polyglot speech synthesis speech is synthesised in multiple languages with the same speaker's voice characteristics [8], [13], [26]. The speaker may have only provided speech training/adaptation data in one language. In such a polyglot speech synthesis system, the voice of someone who speaks only English, for example, can be used to synthesize speech in other languages such as French and German. Secondly, even if a speech synthesis system for a single language is required, for a limited data scenario, the amount of training data for acoustic modeling can be effectively increased by using speech data from multiple speakers in different languages. Lastly, if the amount of data from a new language is limited, the synthesis system can be adapted to the new language by estimating its transform.

This article proposes a new framework, *speaker and language factorization* (SLF), which attempts to factorize speaker-specific/language-specific characteristics in the data. Here, speaker and language transforms are estimated in such a way that each transform is related to only one factor. Ideally, these transforms should be applicable independently, which yields a highly flexible framework for using the transforms. To achieve such "orthogonality", the transforms need to be different in nature from each other.[1] In the proposed SLF framework, the well-known constrained maximum likelihood linear regression (CMLLR) [4] is used for the speaker transforms. For the language transforms, cluster adaptive training

Manuscript received xx, 201x; revised xx, 201x. Part of this work has been presented in Interspeech (Makuhari, Japan, September 2010) [37] and ISCA SSW7 (Kyoto, Japan, September 2010) [39]. The authors are with Toshiba Research Europe Ltd. Cambridge Research Lab, Cambridge CB4 0GZ, United Kingdom. e-mail: heigazen@gmail.com, kate.knill@crl.toshiba.co.uk.

[1]If the transforms are similar in nature, they will subsume the attributes of each other.

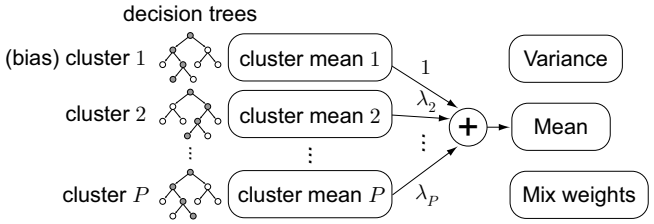

Fig. 1. Cluster adaptive training with cluster-dependent decision trees.

(CAT) [5] is used. Here CAT builds an eigenspace of languages and estimates the position of each language in this space. All clusters are assumed to share the same decision trees in the standard CAT set-up. This is reasonable for speaker or noise modeling, since they are not very sensitive to cluster-dependent context dependency. However, the application to polyglot speech synthesis is more complicated, as the structure of the decision trees associated with each language could be dramatically different. Consider an example where the second and third clusters correspond to tonal (*e.g.*, Mandarin Chinese) and Romance (*e.g.*, French) languages, respectively. Decision tree node splits related to tonal contexts will appear in the trees for the second cluster, whereas they will not for the third cluster. To handle such cluster-specific context-dependency, this article uses CAT with cluster-dependent decision trees [36], [38], which is illustrated in Fig. 1. This is similar to cluster-dependent decision trees used in the additive $\log F_0$ model [38]. However, SLF can be a far more suitable application of this technique as the limitation of sharing decision trees over all languages is expected to be larger.

The remainder of this article is organized as follows: Section II describes the model structure of SLF. Section III gives the training algorithm. Section IV explains the adaptation algorithm. Section V shows experimental results. Concluding remarks and future plans are presented in the final section.

II. Model Structure

Figure 2 shows the block diagram of SLF. This is similar to the structured transform framework [35] to combine CMLLR and CAT. The SLF framework has multiple clusters, each of which can have different decision trees, in contrast to the structured transform framework. The left-hand side of Fig. 2 illustrates the language-adaptation part. The cluster-dependent decision trees are located at the leftmost part of this figure. Cluster mean vectors are associated with the leaf nodes of the cluster-dependent decision trees. Mean vectors in each language-adapted model set are generated by interpolating the cluster mean vectors with language-specific CAT interpolation weights. The generated mean vectors, together with covariance matrices, form the language-adapted model set. Note that decision trees exist for the covariance matrices but these are not shown in the figure. The right-hand side of Fig. 2 illustrates the speaker-adaptation part. In addition to language adaptation by CAT, speaker-specific CMLLR transforms are applied to generate the final speaker- and language-adapted model set.

The emission probability[2] of an observation vector given component, speaker, language, and a set of model parameters can be expressed as

$$p(\boldsymbol{o}(t) \mid m, s, l, \mathcal{M}) = \left|\boldsymbol{A}^{(s)}_{r(m)}\right| \mathcal{N}\left(\boldsymbol{X}^{(s)}_{r(m)}\boldsymbol{\xi}(t)\ ;\ \boldsymbol{M}_m\boldsymbol{\lambda}^{(l)}_{q(m)}, \boldsymbol{\Sigma}_{v(m)}\right), \quad (1)$$

$$\boldsymbol{\lambda}^{(l)}_{q(m)} = \left[1, \lambda^{(l)}_{2,q(m)}, \ldots, \lambda^{(l)}_{P,q(m)}\right]^\top, \quad (2)$$

$$\boldsymbol{X}^{(s)}_{r(m)} = \left[\boldsymbol{b}^{(s)}_{r(m)}, \boldsymbol{A}^{(s)}_{r(m)}\right], \quad (3)$$

$$\boldsymbol{M}_m = \left[\boldsymbol{\mu}_{c(m,1)}, \ldots, \boldsymbol{\mu}_{c(m,P)}\right], \quad (4)$$

$$\boldsymbol{\xi}(t) = \left[1, \boldsymbol{o}(t)^\top\right]^\top, \quad (5)$$

where the following notation will be used in this article.

- $t \in \{1, \ldots, T\}$, $m \in \{1, \ldots, M\}$, $s \in \{1, \ldots, S\}$, $l \in \{1, \ldots, L\}$: frame, Gaussian component, speaker, and language, respectively.
- $q(m) \in \{1, \ldots, Q\}$, $r(m) \in \{1, \ldots, R\}$: CAT and CMLLR regression classes for component m, respectively.
- $v(m) \in \{1, \ldots, V\}$: leaf node for component m in decision trees for the covariance matrices.
- $c(m, i) \in \{1, \ldots, N\}$: leaf node for cluster i of component m in decision trees for cluster mean vectors.
- $T, M, S, L, P, Q, R, V, N$: numbers of frames, Gaussian components, speakers, languages, clusters, CAT and CMLLR regression classes, leaf nodes in decision trees for the covariance matrices, and leaf nodes in decision trees for the cluster mean vectors,[3] respectively.
- $\boldsymbol{o}(t), \boldsymbol{\xi}(t)$: observation vector and extended observation vector at frame t, respectively.
- $\lambda^{(l)}_{i,q}, \boldsymbol{\lambda}^{(l)}_q$: CAT interpolation weight for cluster i and CAT interpolation weight vector, for language l, associated with CAT regression class q, respectively.[4]
- $\boldsymbol{\mu}_n$: cluster mean vector associated with leaf node n.
- $\boldsymbol{M}_m$: matrix of cluster mean vectors for component m.
- $\boldsymbol{A}^{(s)}_r, \boldsymbol{b}^{(s)}_r, \boldsymbol{X}^{(s)}_r$: CMLLR linear transformation matrix, bias vector, and extended transform for speaker s associated with CMLLR regression class r, respectively.
- $\boldsymbol{\Sigma}_k$: covariance matrix associated with leaf node k.
- $\mathcal{M}$: set of model parameters.

The set of model parameters consists of two distinct parts:

1) canonical parameters, $\boldsymbol{\Lambda} = \{\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_k\}$, comprising cluster mean vectors, $\{\boldsymbol{\mu}_n\}$, and covariance matrices, $\{\boldsymbol{\Sigma}_k\}$.[5]

[2]The state-duration probabilities, which are essential in speech synthesis, can also be expressed in the same manner.

[3]This model structure can be interpreted as a tree intersection model, which can effectively represent the vast context space with a small number of parameters [10], [36]. Here N is the total number of leaf nodes of cluster-dependent decision trees and M is the total number of unique combinations of cluster-dependent decision trees, decision trees for covariance matrices, and regression classes for CAT and CMLLR transforms. Thus $M \geq N$, where $M = N$ if all trees are the same.

[4]Here the first cluster is assumed to be a bias cluster, *i.e.*, its weight is fixed to 1.

[5]For this article the estimation of the component prior (mixture weights) and transition matrices are not considered. Their formulae are identical to the standard CAT updates.

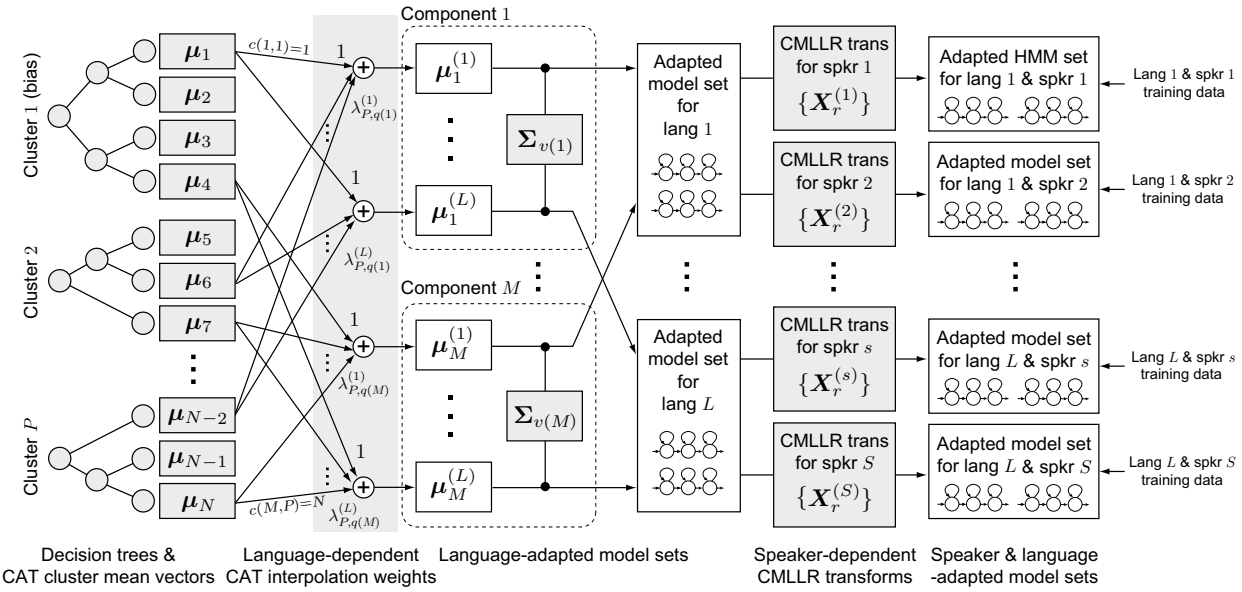

Fig. 2. Block diagram of SLF. Shaded blocks corresponds to the parameters and trees to be updated during the training process.

2) transform parameters, $\boldsymbol{W} = \{\boldsymbol{X}_r^{(s)}, \boldsymbol{\lambda}_q^{(l)}\}$, comprising speaker-specific CMLLR transforms, $\{\boldsymbol{X}_r^{(s)}\}$ and language-specific CAT interpolation weight vectors, $\{\boldsymbol{\lambda}_q^{(l)}\}$.

Thus $\mathcal{M} = \{\boldsymbol{\Lambda}, \boldsymbol{W}\}$. The next section describes how to train these parameters.

III. Training

A. Auxiliary function

The goal is to estimate the parameters that maximize the log likelihood given the training data with its associated transcriptions and speaker/language labels. Like speaker adaptive training (SAT), the expectation-maximization (EM) algorithm is used. An iterative approach is adopted where first the transform parameters are estimated, then the canonical parameters. The whole process is then repeated.

From Eq. (1), the auxiliary function of the EM algorithm is given as

$$\mathcal{Q}(\mathcal{M}; \hat{\mathcal{M}}) = -\frac{1}{2} \sum_{m,t,s,l} \gamma_m(t,s,l) \left\{ \left(\hat{\boldsymbol{o}}_{r(m)}^{(s)}(t) - \boldsymbol{\mu}_m^{(l)} \right)^\top \boldsymbol{\Sigma}_{v(m)}^{-1} \left(\hat{\boldsymbol{o}}_{r(m)}^{(s)}(t) - \boldsymbol{\mu}_m^{(l)} \right) + \log \left| \boldsymbol{\Sigma}_{v(m)} \right| - 2 \log \left| \boldsymbol{A}_{r(m)}^{(s)} \right| \right\} + C, \quad (6)$$

where C is a constant independent of $\mathcal{M}$, $\hat{\mathcal{M}}$ is the current estimate of the set of model parameters, and $\gamma_m(t,s,l)$ is the posterior probability of component m generating $\boldsymbol{o}(t)$ given s and l, calculated using the forward-backward algorithm with $\hat{\mathcal{M}}$. $\hat{\boldsymbol{o}}_{r(m)}^{(s)}(t)$ and $\boldsymbol{\mu}_m^{(l)}$ correspond to the transformed observation vector and the interpolated cluster mean vectors given as

$$\hat{\boldsymbol{o}}_{r(m)}^{(s)}(t) = \boldsymbol{X}_{r(m)}^{(s)} \boldsymbol{\xi}(t), \quad (7)$$

$$\boldsymbol{\mu}_m^{(l)} = \boldsymbol{M}_m \boldsymbol{\lambda}_{q(m)}^{(l)}. \quad (8)$$

B. Canonical parameter re-estimation

Substituting Eqs. (2), (4), and (8) into Eq. (6) yields[6]

$$\mathcal{Q}(\mathcal{M}; \hat{\mathcal{M}}) = -\frac{1}{2} \sum_{m,t,s,l} \gamma_m(t,s,l) \left(\sum_{i,j} \boldsymbol{\mu}_{c(m,i)}^\top \lambda_{i,q(m)}^{(l)} \boldsymbol{\Sigma}_{v(m)}^{-1} \lambda_{j,q(m)}^{(l)} \boldsymbol{\mu}_{c(m,j)} - 2 \sum_i \boldsymbol{\mu}_{c(m,i)}^\top \lambda_{i,q(m)}^{(l)} \boldsymbol{\Sigma}_{v(m)}^{-1} \hat{\boldsymbol{o}}_{r(m)}^{(s)}(t) \right) \quad (9)$$

$$= -\frac{1}{2} \sum_{m,i} \left(\boldsymbol{\mu}_{c(m,i)}^\top \boldsymbol{G}_{ii}^{(m)} \boldsymbol{\mu}_{c(m,i)} + 2 \sum_{j \neq i} \boldsymbol{\mu}_{c(m,i)}^\top \boldsymbol{G}_{ij}^{(m)} \boldsymbol{\mu}_{c(m,j)} - 2 \boldsymbol{\mu}_{c(m,i)}^\top \boldsymbol{k}_i^{(m)} \right), \quad (10)$$

where $\boldsymbol{G}_{ij}^{(m)}$ and $\boldsymbol{k}_i^{(m)}$ are accumulated statistics defined as

$$\boldsymbol{G}_{ij}^{(m)} = \sum_{t,s,l} \gamma_m(t,s,l) \lambda_{i,q(m)}^{(l)} \boldsymbol{\Sigma}_{v(m)}^{-1} \lambda_{j,q(m)}^{(l)}, \quad (11)$$

$$\boldsymbol{k}_i^{(m)} = \sum_{t,s,l} \gamma_m(t,s,l) \lambda_{i,q(m)}^{(l)} \boldsymbol{\Sigma}_{v(m)}^{-1} \hat{\boldsymbol{o}}_{r(m)}^{(s)}(t). \quad (12)$$

Using this auxiliary function the ML estimates of the canonical parameter $\boldsymbol{\Lambda}$ can be found. Initially just the cluster mean

[6]Constant terms independent of the cluster mean vectors are omitted.

vectors are considered. The first partial derivative of Eq. (10) with respect to $\boldsymbol{\mu}_n$ is given by

$$\frac{\partial \mathcal{Q}(\mathcal{M};\hat{\mathcal{M}})}{\partial \boldsymbol{\mu}_n} = \boldsymbol{k}_n - \boldsymbol{G}_{nn}\boldsymbol{\mu}_n - \sum_{\nu \neq n} \boldsymbol{G}_{n\nu}\boldsymbol{\mu}_\nu, \qquad (13)$$

where

$$\boldsymbol{G}_{n\nu} = \sum_{\substack{m,i,j \\ c(m,i)=n \\ c(m,j)=\nu}} \boldsymbol{G}_{ij}^{(m)}, \qquad \boldsymbol{k}_n = \sum_{\substack{m,i \\ c(m,i)=n}} \boldsymbol{k}_i^{(m)}. \qquad (14)$$

By setting Eq. (13) to $\mathbf{0}$, the ML estimate of $\boldsymbol{\mu}_n$ can be determined as

$$\hat{\boldsymbol{\mu}}_n = \boldsymbol{G}_{nn}^{-1}\left(\boldsymbol{k}_n - \sum_{\nu \neq n} \boldsymbol{G}_{n\nu}\boldsymbol{\mu}_\nu\right). \qquad (15)$$

It can be seen from Eq. (15) that the ML estimate of $\boldsymbol{\mu}_n$ depends on all other cluster mean vectors. In principle, the optimization should therefore be repeated over all cluster mean vectors until they converge. Alternatively, all cluster mean vectors can be determined simultaneously by solving the following set of linear equations:[7]

$$\begin{bmatrix} \boldsymbol{G}_{11} & \cdots & \boldsymbol{G}_{1N} \\ \vdots & \ddots & \vdots \\ \boldsymbol{G}_{N1} & \cdots & \boldsymbol{G}_{NN} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\mu}}_1 \\ \vdots \\ \hat{\boldsymbol{\mu}}_N \end{bmatrix} = \begin{bmatrix} \boldsymbol{k}_1 \\ \vdots \\ \boldsymbol{k}_N \end{bmatrix}. \qquad (16)$$

Although the dimensionality of Eq. (16) can be hundreds of thousands, it is sparse.[8] Therefore, it can be stored and solved efficiently using a sparse matrix storage and solver.

By taking the first partial derivative of Eq. (6) with respect to $\boldsymbol{\Sigma}_k$ and setting it to $\mathbf{0}$, the ML estimate of the covariance matrices can be determined as

$$\hat{\boldsymbol{\Sigma}}_k = \frac{\sum_{\substack{t,s,l,m \\ v(m)=k}} \gamma_m(t,s,l)\bar{\boldsymbol{o}}_{r(m)}^{(s)}(t)\,\bar{\boldsymbol{o}}_{r(m)}^{(s)}(t)^\top}{\sum_{\substack{t,s,l,m \\ v(m)=k}} \gamma_m(t,s,l)}, \qquad (17)$$

where

$$\bar{\boldsymbol{o}}_{r(m)}^{(s)}(t) = \hat{\boldsymbol{o}}_{r(m)}^{(s)}(t) - \boldsymbol{\mu}_m^{(l)}. \qquad (18)$$

C. Transform parameter re-estimation

Reestimation of the transform parameters is a simple iterative process. Given the CAT interpolation weight vectors, $\{\boldsymbol{\lambda}_q^{(l)}\}$, the adapted mean vectors, $\{\boldsymbol{\mu}_m^{(l)}\}$, are used to estimate the CMLLR transforms, $\{\boldsymbol{X}_r^{(s)}\}$, as described in [5]. Then given the CMLLR transforms, $\{\boldsymbol{X}_r^{(s)}\}$, the CAT interpolation weight vectors, $\{\boldsymbol{\lambda}_q^{(l)}\}$, are estimated using the transformed feature vectors, $\{\hat{\boldsymbol{o}}_r^{(s)}(t)\}$, as described in [4], [20].

[7] If all covariance matrices are diagonal, each dimension of $\{\boldsymbol{\mu}_n\}$ can be determined independently.

[8] $\boldsymbol{G}_{n\nu} \neq \mathbf{0}$ only if n-th and ν-th nodes appear simultaneously in the training data. Due to the nature of decision trees (hard split of data), most combinations do not appear in the training data.

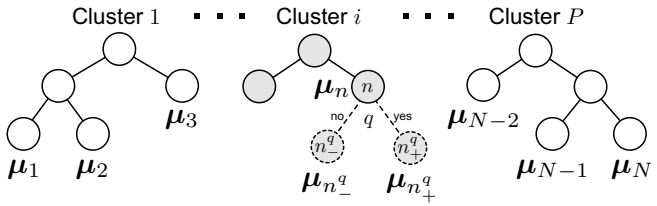

Fig. 3. Overview of tree-reconstruction process. Node n is associated with the decision tree for cluster i. Shaded parts correspond to parameters and a tree to be updated.

D. Tree reconstruction

The conventional cluster-based techniques such as eigenvoice [7] and CAT [5] assume that all clusters have the same parameter tying structure, *i.e.*, the same decision trees. However, this restriction is not inherent to these techniques: each cluster can in fact have its own parameter tying structure. Recently, cluster-based techniques with cluster-dependent decision trees have been proposed [14], [38], where different decision trees are built for each cluster. In these techniques, the cluster-dependent decision trees are expected to capture cluster-specific context dependency.

As building multiple trees simultaneously [38] is computationally expensive, an iterative, cluster-by-cluster reconstruction approach [14] is used. While reconstructing decision trees for a cluster, the parameters of all other clusters, which include the structure of the other trees, their associated cluster mean vectors, covariance matrices, and transform parameters, are fixed.[9] The goal is to build decision trees and estimate associated parameters that maximize the log likelihood given the training data, while maintaining the balance between model complexity and accuracy.

As illustrated in Fig. 3, let us consider the situation that node n associated with the decision tree for cluster i is divided into two new terminal nodes, n_+^q and n_-^q, by question q. By applying the assumptions introduced in [11], the total log likelihood of node n for cluster i can be calculated as[10]

$$\mathcal{L}(n) = -\frac{1}{2}\sum_{m \in \mathcal{S}(n)} \Bigg(\boldsymbol{\mu}_{c(m,i)}^\top \boldsymbol{G}_{ii}^{(m)} \boldsymbol{\mu}_{c(m,i)} + 2\sum_{j \neq i} \boldsymbol{\mu}_{c(m,i)}^\top \boldsymbol{G}_{ij}^{(m)} \boldsymbol{\mu}_{c(m,j)} - 2\boldsymbol{\mu}_{c(m,i)}^\top \boldsymbol{k}_i^{(m)} \Bigg), \qquad (19)$$

where $\mathcal{S}(n)$ denotes a set of components associated with node n. Because all cluster mean vectors associated with node n will be tied ($\forall_{m \in \mathcal{S}(n)} \boldsymbol{\mu}_{c(m,i)} = \boldsymbol{\mu}_n$), Eq. (19) can be re-written as

$$\mathcal{L}(n) = -\frac{1}{2}\boldsymbol{\mu}_n^\top \left(\sum_{m \in \mathcal{S}(n)} \boldsymbol{G}_{ii}^{(m)} \right) \boldsymbol{\mu}_n + \boldsymbol{\mu}_n^\top \sum_{m \in \mathcal{S}(n)} \left(\boldsymbol{k}_i^{(m)} - \sum_{j \neq i} \boldsymbol{G}_{ij}^{(m)} \boldsymbol{\mu}_{c(m,j)} \right). \qquad (20)$$

[9] Here it is assumed that no cluster mean vectors are shared across clusters.

[10] Constant terms independent of the cluster mean vectors are omitted.

If all the canonical parameters associated with the other trees and all the transform parameters are assumed to be unchanged, the ML estimates of $\boldsymbol{\mu}_n$ can be determined as

$$\hat{\boldsymbol{\mu}}_n = \left(\sum_{m \in \mathcal{S}(n)} \boldsymbol{G}_{ii}^{(m)} \right)^{-1} \sum_{m \in \mathcal{S}(n)} \left(\boldsymbol{k}_i^{(m)} - \sum_{j \neq i} \boldsymbol{G}_{ij}^{(m)} \boldsymbol{\mu}_{c(m,j)} \right). \quad (21)$$

Substituting Eq. (21) into Eq. (20) yields

$$\mathcal{L}(n) = \frac{1}{2} \hat{\boldsymbol{\mu}}_n^\top \left(\sum_{m \in \mathcal{S}(n)} \boldsymbol{G}_{ii}^{(m)} \right) \hat{\boldsymbol{\mu}}_n. \quad (22)$$

The log-likelihood gain which results from splitting node n into nodes n_+^q and n_-^q with question q is computed as

$$\Delta\mathcal{L}(n; q) = \mathcal{L}(n_+^q) + \mathcal{L}(n_-^q) - \mathcal{L}(n). \quad (23)$$

Based on the log-likelihood gain, the best question to split node n can be selected as

$$\hat{q}_n = \arg\max_q \Delta\mathcal{L}(n; q). \quad (24)$$

By repeating this process from the root node until a stopping criterion is met, this decision tree can be re-constructed. Splitting can be stopped according to the log-likelihood gain by a heuristic threshold [11], cross validation [19], or an information criterion such as the minimum description length (MDL) criterion [18]. After re-constructing decision trees for a cluster, decision trees for the next cluster are re-built in the same manner. This process is repeated from cluster 1 to P.

Decision trees for the covariance matrices and regression classes for the CMLLR transforms and the CAT interpolation weight vectors are also required. In the experiments which will be described in Section V, covariance matrices (and component priors) were clustered together with the bias cluster. Furthermore, regression classes for CMLLR transforms and CAT interpolation weight vectors were defined globally (silence, pause, and others).[11]

E. Initialization

While training a model using the EM algorithm, initialization is always an important issue. There exist several possible ways of initializing the parameters of an SLF model. One option is to initialize the parameters with a speaker-adaptively trained language-independent (LI-SAT) model.

First an LI-SAT model is trained in the standard SAT manner using the training data from multiple speakers in different languages. Then, an SLF model is initialized using this LI-SAT model as follows: The number of clusters P is set to $L + 1$. The decision trees for cluster 1 (bias cluster) and their associated cluster mean vectors are initialized to those of the LI-SAT model. The covariance matrices, space weights for multi-space probability distributions (MSD) [23], and their parameter sharing structure are also initialized to those of the LI-SAT model. A specific language tag is assigned to each of $2, \ldots, P$ clusters, *e.g.*, clusters 2, 3, and 4 are for German, Spanish, and French, respectively. The decision trees for clusters $2, \ldots, P$ are initialized to have only root nodes, and the cluster mean vectors associated with these root nodes are set to $\mathbf{0}$. A set of CAT interpolation weights are simply set to 1 or 0 according to their assigned language tags[12] as

$$\lambda_{i,q}^{(l)} = \begin{cases} 1 & i = 1 \text{ or its language tag is } l \\ 0 & \text{Otherwise} \end{cases}.$$

A set of CMLLR transforms of the LI-SAT model are used to initialize that of the SLF model. As a result, the SLF model which gives exactly the same log likelihood on the training data as the LI-SAT model is achieved.

From this initial stage, the process of training the SLF model is an interleaving update process as described below:

1) Initialize canonical parameters $\hat{\boldsymbol{\Lambda}}_0 = \{\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_k\}$ and transform parameters $\hat{\boldsymbol{W}}_0 = \{\boldsymbol{X}_r^{(s)}, \boldsymbol{\lambda}_q^{(l)}\}$, set $j = 0$.
2) Re-construct decision trees cluster-by-cluster from cluster 1 to P.[13]
3) Estimate $\hat{\boldsymbol{\Lambda}}_{j+1}$ given $\hat{\boldsymbol{\Lambda}}_j$ and $\hat{\boldsymbol{W}}_j$.
4) Estimate $\hat{\boldsymbol{W}}_{j+1}$ given $\hat{\boldsymbol{\Lambda}}_{j+1}$ and $\hat{\boldsymbol{W}}_j$.
5) $j = j + 1$. Go to 2) until convergence.

IV. Adaptation to Target Condition

Adaptation to a target condition, which is a particular pair of speaker and language, involves two distinct sub-steps of estimating speaker-specific CMLLR transforms and language-specific CAT interpolation weight vectors, $\{\boldsymbol{X}_r^{(s)}, \boldsymbol{\lambda}_q^{(l)}\}$, given the set of canonical model parameters, $\boldsymbol{\Lambda}$, similar to the transform parameter estimation described in Section III-C. These transforms are used to construct the adapted model for synthesis.

A. Intra-lingual speaker adaptation

Intra-lingual speaker adaptation is straightforward. Given the adaptation data from the target speaker in one of the training languages, only the speaker transform, $\{\boldsymbol{X}_r^{(s)}\}$, of the pair of speaker and language transforms, $\{\boldsymbol{X}_r^{(s)}, \boldsymbol{\lambda}_q^{(l)}\}$, is estimated [4], [20] as $\{\boldsymbol{\lambda}_q^{(l)}\}$ can be set to the one estimated in the training process.

[11]Liang and Dines reported that the use of a global transform worked better than many regression tree-clustered transforms in cross-lingua speaker adaptation [9]. Therefore, the three simple regression classes were used here. Increasing the number of regression classes is straightforward but not investigated.

[12]Other initialization schemes are also possible, such as random or eigenvoice-style initialization. The deterministic and binary initialization scheme, which was used in the experiments, requires $L + 1$ clusters as it creates a separate cluster for each of the languages. However, with other initialization approaches, having $L + 1$ clusters is not strictly necessary. This will be preferred if there are a large number of languages in the training data. A preliminary experiment showed no significant difference between the binary and random initialization approaches.

[13]Thus results depend on the order of re-construction of decision trees.

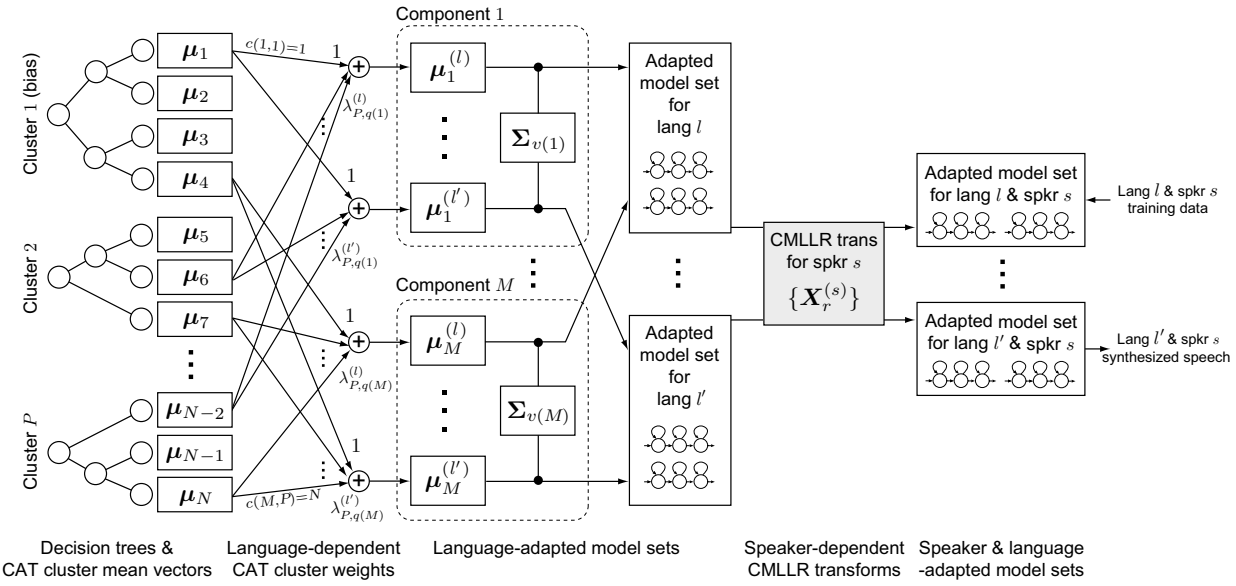

Fig. 4. Block diagram of cross-lingual speaker adaptation in SLF. Shaded blocks correspond to those being updated.

B. Cross-lingual speaker adaptation for polyglot synthesis

Figure 4 illustrates cross-lingual speaker adaptation (CLSA) in the SLF framework. First $\{\boldsymbol{X}_r^{(s)}\}$ is estimated in the same way as intra-lingual speaker adaptation. Then $\{\boldsymbol{X}_r^{(s)}, \boldsymbol{\lambda}_q^{(l')}\}$ can be obtained by combining $\{\boldsymbol{X}_r^{(s)}\}$ with $\{\boldsymbol{\lambda}_q^{(l')}\}$ for any of the training languages. From the adapted model, speech in language l' with the voice characteristics of speaker s can be synthesized. As a result, the SLF framework can perform polyglot speech synthesis.

C. Language adaptation

Language adaptation aims to adapt an existing SLF model to a new language. Figure 5 illustrates the adaptation process. It is described as follows:

1) Initialize $\{\boldsymbol{X}_r^{(s')}, \boldsymbol{\lambda}_q^{(l'')}\}, \ldots, \{\boldsymbol{X}_r^{(s'')}, \boldsymbol{\lambda}_q^{(l'')}\}$ for speakers/language pairs in the adaptation data, where l'' denotes the new language and $\{s', \ldots, s''\}$ correspond to the speakers in the adaptation data.[14]
2) Reestimate $\{\boldsymbol{X}_r^{(s')}\}, \ldots, \{\boldsymbol{X}_r^{(s'')}\}$ given $\{\boldsymbol{\lambda}_q^{(l'')}\}$ and $\boldsymbol{\Lambda}$ [4], [20].
3) Reestimate $\{\boldsymbol{\lambda}_q^{(l'')}\}$ given $\{\boldsymbol{X}_r^{(s')}\}, \ldots, \{\boldsymbol{X}_r^{(s'')}\}$ and $\boldsymbol{\Lambda}$ [5].
4) Go to 2) until convergence.
5) The number of clusters is increased from P to $P+1$. Decision trees for cluster $P+1$ are initialized to have only root nodes. All cluster mean vectors associated with the root nodes of the decision trees for cluster $P+1$ are set to $\mathbf{0}$. The CAT interpolation weights for cluster $P+1$ are set to 1.
6) Accumulate statistics then build decision trees for cluster $P+1$.

[14]In the experiments reported in Section V, $\{\boldsymbol{X}_r^{(s')}\}, \ldots, \{\boldsymbol{X}_r^{(s'')}\}$ were initialized to an identity transform and $\{\boldsymbol{\lambda}_q^{(l'')}\}$ was initialized to the one estimated from all training data.

It is possible to perform language adaptation omitting steps 5) and 6). This has the advantage that building the decision trees for the target language is not required. However, where the target language is not well represented by the training languages this may impact synthesis performance. This may occur often, given the wide range of variations in languages. Thus it is likely that the context-dependency in the target language cannot be fully expressed by a point (or a set of points) in the language eigenspace. In this case, having target language-specific decision trees (steps 5) and 6)), provides an ability to capture the target language-specific context-dependency. This is a new and powerful way to do adaptation to a target condition.

The process of adding new trees is similar to the tree reconstruction described in Section III-D. It is interesting to note that incrementally adding a new language to an existing system using language adaptation can be viewed as an approximation of full SLF training. It allows a new system to be built from an existing system by having additional decision trees for the new data. This eliminates the requirement to store/access speech data used for training the existing system while building a new system.

Note that the CAT interpolation weights for the additional cluster are fixed to 1 in Fig. 5 and the experiment reported in Section V-E, as there is an arbitrary scaling between additional cluster mean vectors and their interpolation weights. Fixing $\forall_q \lambda_{P+1,q}^{(l')} = 1$ removes this issue.

V. EXPERIMENTS

A. Data preparation

A range of multilingual speech databases are available, such as GlobalPhone [16]. However, none of them are designed for speech synthesis purposes. Therefore, a new database was recorded. The database consisted of five languages; North American (US) English, British (UK) English, European Spanish, European French, and Standard German. There were 10

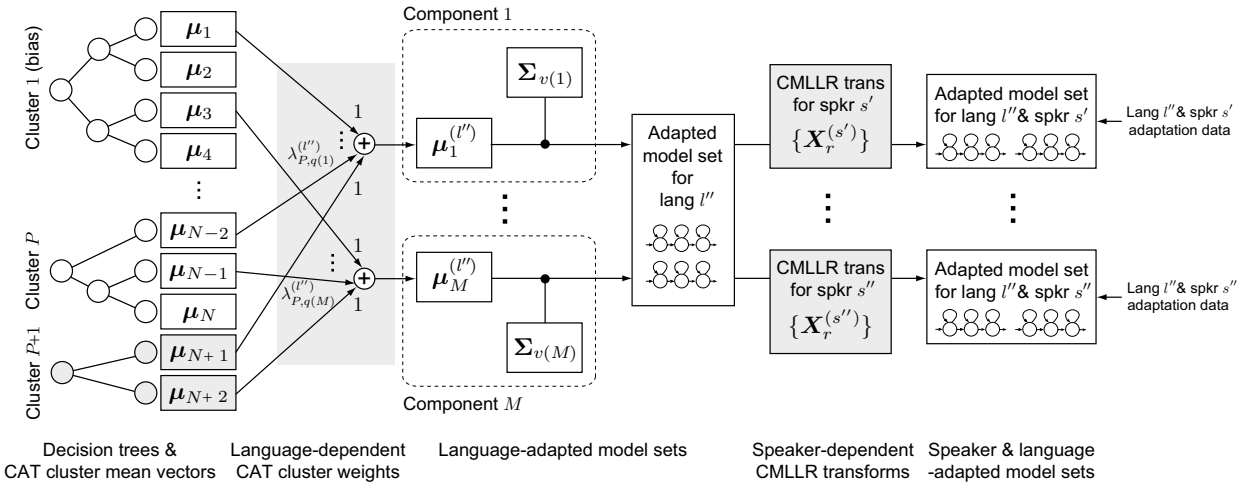

Fig. 5. Block diagram of language adaptation. Language-specific CAT interpolation weights for the target language are estimated and additional cluster-dependent decision trees, which represent the unique context-dependency of the target language, are built. Shaded blocks correspond to those being updated.

non-professional speakers (five male and five female) in each language. These speakers were selected from four age ranges (1 from 13–18, 2 from 20–30, 1 from 30–50, and 1 from 50–70) for each gender. Speakers did not have strong regional accents. Each speaker uttered the same 50 phonetically rich sentences which covered all phones in the language and another set of 50–100 which were selected from various domains (and differed among speakers). The total recording duration for each speaker was between eight and fifteen minutes. A headset microphone was used to record the voices. All recordings were in a standard recording room with low reverberation and minimal background noise. To avoid the effect of variations in recording conditions, the same microphone and recording room were used for all speakers. The sampling frequency was 48 kHz, later down-sampled to 16 kHz. These recordings were used for these experiments.

A universal phone set, which covered all training languages, was defined and used. Each phone symbol in this phone set has an equivalent transcription in the IPA alphabet [1]. The recording scripts were automatically converted into the corresponding phone sequences using a proprietary text analysis engine. A proprietary HMM-based automatic aligner was then used to extract the phone segmentations. A universal context-dependent label format, covering possible contexts in the training languages, was also defined. The contexts used in this format were similar to those in [25]: they included phonetic, prosodic, and grammatical contexts. The fundamental frequency (F_0) values of the recordings were automatically extracted from the recordings using the voting method [33].

B. Experimental setup

The speech analysis conditions and model topologies used in this experiment were the same as those of HTS 2008 [33], except the use of 23 Bark-scale band aperiodicities [32] rather than 5-band ones. Refer to [33] for details. Five speaker-adaptively trained language-dependent (LD-SAT) models and a language-independent (LI-SAT) model were trained. The LI-SAT model was trained using the data from all training speakers and in all training languages, while each of the LD-SAT models was trained using the data from all training speakers in one language. This LI-SAT model was used to initialize the SLF model. To store and solve the sparse set of linear equations of Eq. (16), the compressed sparse row (CSR) format and the parallel sparse direct linear solver (PARDISO) [15] were used, respectively. Decision tree-based context clustering based on 10-fold cross validation [19] was used to improve the robustness of the constructed trees.[15] Five iterations of SLF tree update and model estimation were run. All adaptation was performed in a supervised, batch adaptation mode. After training the models, speech parameters for the test sentences were generated from the models using the speech parameter generation algorithm including a global variance (GV) term [22]. For each target speaker, a context-independent Gaussian distribution with a diagonal covariance matrix, which modeled the probability distribution of GVs, was estimated from the adaptation data. From the generated speech parameters, speech waveforms were synthesized using the source-filter model.

A variety of subjective listening tests were conducted. All subjective listening tests were crowd-sourced.[16] To avoid non-native speakers participating in the evaluation, only subjects who lived in the home country of each language (German→Germany, UK English→United Kingdom, US English→United States, Spanish→Spain, and French→France), could participate in the evaluation.

All paired-comparison preference listening tests reported here compared the naturalness of synthesized speech. To ensure that pairs of speech samples were played equally often in AB as in BA order, both orders were regarded as different pairs. Pairs of samples were randomly chosen and presented for each subject. After listening to each pair of samples, the subjects were asked to choose their preferred one. Note

[15]The LI-SAT and LD-SAT models were also trained with the same setting for decision tree-based context clustering.

[16]Amazon Mechanical Turk (http://www.mturk.com/) was used for experiments in German, UK English, and US English. Clickworker (http://www.clickworker.com/) was used for experiments in French and Spanish.

TABLE I
NUMBERS OF LEAF NODES AND PARAMETERS FOR MEL-CEPSTRAL COEFFICIENTS, $\log F_0$, BAND APERIODICITY, AND STATE DURATIONS IN THE SLF MODEL.

Cluster	Speech parameter			
	mel-cep.	$\log F_0$	band ap.	dur.
1 (bias)	2 071	4 059	5 940	1 168
2	102	3 304	20	46
3	164	3 744	17	38
4	88	3 582	18	27
5	129	3 259	25	21
6	125	2 956	28	41
Before intersection (N)	2 679	20 904	6 048	1 341
After intersection (M)	172 135	607 915	8 039	21 457
# of parameters	570 000	45 257	827 172	12 545
Total # of parameters	1 454 974			

TABLE II
THE VALUES OF THE ESTIMATED LANGUAGE-SPECIFIC CAT INTERPOLATION WEIGHTS FOR THE TRAINING LANGUAGES FOR MEL-CEPSTRAL COEFFICIENTS AND $\log F_0$ VALUES. THE LARGEST WEIGHTS AMONGST CLUSTERS IN EACH LANGUAGE ARE IN THE BOLD FONT.

Speech parameter	Language	Cluster				
		2	3	4	5	6
mel-cep.	German	**.619**	.401	.002	.340	.325
	UK English	.294	**.575**	.424	.252	.233
	US English	.339	.457	**.846**	.255	.236
	Spanish	.489	.381	.048	**.627**	.398
	French	.428	.314	.070	.383	**.682**
$\log F_0$	German	**.897**	.049	.136	.102	.098
	UK English	.037	**.879**	.184	.057	.084
	US English	.111	.197	**.821**	.043	.091
	Spanish	.064	.118	.117	**.909**	.084
	French	.065	.047	.174	.092	**.914**

TABLE III
PREFERENCE SCORES (%) OF SPEECH SYNTHESIZED FROM THE LD-SAT (LD), LI-SAT (LI), AND SLF (SLF) MODELS. NOTE THAT N/P DENOTES "NO PREFERENCE." SCORES WITH STATISTICALLY SIGNIFICANT PREFERENCE AT $p < 0.05$ LEVEL ARE IN THE BOLD FONT.

Language	Preference score				p (t-test)
	LD	LI	SLF	N/P	
German	39.7	36.2	–	24.1	0.164
	35.2	–	**46.8**	18.0	0.001
	–	33.8	**43.2**	23.0	0.005
UK English	**44.8**	37.5	–	17.7	0.023
	36.1	–	**48.6**	15.3	<0.0001
	–	31.8	**50.8**	17.4	<0.0001
US English	29.1	**55.3**	–	15.6	<0.0001
	26.2	–	**60.6**	13.1	<0.0001
	–	36.7	**47.6**	15.6	0.002
Spanish	**44.3**	32.3	–	23.3	0.001
	39.4	–	41.9	18.7	0.249
	–	28.1	**48.1**	23.8	<0.0001
French	37.8	42.5	–	19.7	0.110
	37.2	–	**46.5**	16.4	0.007
	–	34.7	**43.0**	22.6	0.010

that the subjects could select "No preference" if they had no preference. The metrics to exclude cheats (preference for the second sample and deviation in system preference) in [3] were used while computing preference scores.

All mean opinion score (MOS) tests and differential MOS (DMOS) tests reported here evaluated the naturalness of synthesized speech and the speaker similarity, respectively. Test samples were randomly chosen and presented for each subject. In the MOS test, after the subjects had listened to a test sample, they were asked to assign it a naturalness score from the five-point Likert scale (5: completely natural – 1: completely unnatural). In the DMOS test, after the subjects had listened to the target speaker's natural speech and a test sample (same sentence), they were asked to assign it a similarity score from the five-point Likert scale (5: exactly the same – 1: completely different).

C. Building single language systems

The first experiment evaluated the performance of SLF in building speech synthesis systems for single languages. Training data consisted of 4 631 utterances by 40 speakers in five languages (German, UK and US English, Spanish, and French). Eight (four female and four male) speakers per language were used for training.[17] The remaining two (one female and one male) speakers per language were used for evaluation. A hundred utterances not included in the training data were used for estimating speaker transforms, and fifty sentences included in neither training nor adaptation data were used for evaluation.

Table I shows the numbers of leaf nodes and parameters for spectrum (mel-cepstral coefficients), $\log F_0$, excitation (band aperiodicity), and state durations in the SLF model. Note that the total numbers of parameters of the LD-SAT and LI-SAT models were comparable to that of the SLF model; there were 1 695 011 parameters in the five LD-SAT models, and the LI-SAT model had 1 432 941 parameters. It can be seen from the table that the numbers of leaf nodes assigned to $\log F_0$ for the non-bias clusters were comparable. However for mel-cepstral coefficients, band aperiodicities, and durations there were far fewer language-dependent leaf nodes. Table II shows the estimated language-specific CAT interpolation weights for the training languages. Note that cluster 1 was omitted as its weights were fixed to 1. It can be seen from the table that the second, third, fourth, fifth, and sixth clusters tended to represent German, UK English, US English, Spanish, and French, respectively. This tendency was due to the deterministic, binary initialization described in Section III-E. For mel-cepstral coefficients, the non-bias clusters were shared across languages; for example, the second cluster made the largest contribution to German but it also contributed to Spanish and French. The fourth cluster made large contributions to UK and US English but almost no contribution to other languages. This suggests that this cluster represents the common factors between UK and US English. On the other hand, weights for $\log F_0$ were almost binary. For example, the sixth cluster made a large contribution to French but almost no contribution to the other languages. These results are intuitive because $\log F_0$ has a large language-dependency whereas the other parameters have a large dependency on language-common factors such as phone classes.

Five paired-comparison preference listening tests were conducted. These tests compared synthesized speech generated from LD-SAT, LI-SAT, and SLF models over 100 (2 speakers

[17]The total duration of the training data was about seven hours long.

$\times$ 50 sentences) evaluation utterances. One subject could evaluate a maximum of 40 pairs. Each pair was evaluated by four subjects. Table III shows the preference test results. It can be seen from the table that SLF achieved the best preference scores among the three systems in all languages. An informal analysis of the synthesized speech showed that the LD-SAT models could produce more natural prosody than the LI-SAT model but their segmental quality sometimes degraded due to data sparseness, whereas the LI-SAT model produced flat prosody but its segmental quality was better than those of the LD-SAT models. The analysis also showed that the segmental quality of the SLF model was similar to that of the LI-SAT model but its prosody was similar to that of the LD-SAT models. These results indicate that even when building a synthesis system for a single language, the use of SLF to take data from multiple languages is advantageous as it can effectively increase the amount of data for training the acoustic models.

D. Polyglot speech synthesis

The second experiment evaluated the naturalness and the speaker similarity of synthetic speech in polyglot speech synthesis. The LD-SAT, LI-SAT, and SLF models from the previous section were used. Six German-English bilingual speakers (three female speakers GF1–GF3 and three male speakers GM1–GM3) from the EMIME German-English bilingual database [28] were used for adaptation. This database was processed (segmentation, text analysis, feature extraction) in the same manner as the database used for training. The adaptation data consisted of 99 utterances for each target speaker. Forty-six sentences included in neither the training nor adaptation data were used for evaluation.

A mean opinion score (MOS) test and a differential MOS (DMOS) test were conducted. The source language was German and the target language was English.[18] The speech samples to be evaluated were synthesized from the systems below:

1) US English LD-SAT model without adaptation (AVM).
2) US English LD-SAT model adapted with CMLLR transforms for the target speaker estimated from the German adaptation data using the state-mapping CLSA method based on transform mapping[19] [13], [29] (CROSS).
3) LI-SAT model adapted with CMLLR transforms for the target speaker estimated from the German adaptation data (LI-SAT).
4) SLF model adapted with CMLLR transforms for the target speaker estimated from the German adaptation data and the pre-estimated CAT interpolation weights for US English (SLF).

[18]According to personal communication with Dr. Mirjam Wester of University of Edinburgh, who developed the bilingual database, many of the speakers in the database have mixed accents of English. The dominant accents in their speech were GF1) US, GF2) UK/German, GF3) German/US, GM1) US, GM2) German/UK, and GM3) UK/German.

[19]The state-mapping CLSA method based on transform mapping implemented in HTS-2.2β was used. Although the authors also investigated the performance of the CLSA method based on data mapping [29], no statistically significant difference was observed.

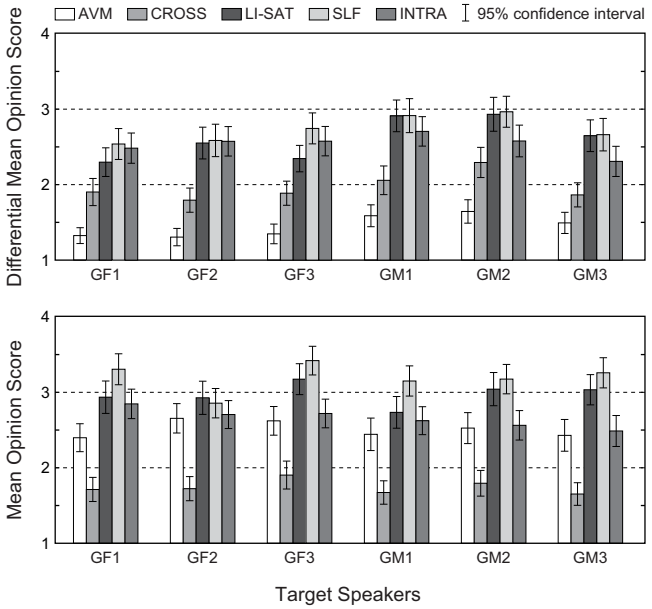

Fig. 6. Differential MOS and MOS test results of synthesized speech from the different adapted models.

5) US English LD-SAT model adapted with CMLLR transforms for the target speaker estimated from target speaker's English adaptation data (INTRA).

In addition to the above samples, vocoded natural speech samples were included in the experiment. The naturalness and similarity scores of the vocoded natural speech were around 4.7 for all the target speakers. System 2) was based on the conventional state-mapping-based CLSA method [9], [13], [29]. System 4) can be viewed as a speaker adaptively trained version of HMM-based polyglot speech synthesis based on mixing mono-lingual corpora [8] System 5) used only US English data for both training and adaptation. There were 2 208 (6 speakers $\times$ 46 sentences $\times$ 8 systems) samples in the test. One subject could evaluate a maximum of 40 and 80 test samples in the DMOS and MOS tests, respectively. Each test sample was evaluated by three subjects.

Figure 6 shows the experimental results. It can be seen from the figure that all the adaptation techniques achieved better similarity than AVM. There were significant differences between CROSS and LI-SAT/SLF in speaker similarity. It is known that adaptation performance of the state-mapping CLSA method severely degrades if there is a large mismatch between the acoustic models for the source and target languages [9], [12], [29]. This mismatch can be caused by inconsistencies in the training data for the source and target languages, such as speaker variations, recording conditions, and amount of data. On the other hand, as LI-SAT and SLF use all languages together while estimating the models, they are less affected by the mismatches between the source and target languages. Furthermore, SLF achieved the same or slightly better similarity as INTRA. This indicates that the speaker and language factors were successfully factorized by the SLF framework. However, there still exists a large gap in speaker similarity between synthesized and natural speech.

TABLE IV
PREFERENCE SCORES (%) OF SPEECH SYNTHESIZED FROM THE SLF MODEL WITHOUT LANGUAGE ADAPTATION (NA), WITH ESTIMATED CAT INTERPOLATION WEIGHTS (W), WITH ESTIMATED CAT INTERPOLATION WEIGHTS AND ADDITIONAL DECISION TREES (W+T). NOTE THAT N/P DENOTES "NO PREFERENCE." SCORES WITH STATISTICALLY SIGNIFICANT PREFERENCE AT $p < 0.05$ LEVEL ARE IN THE BOLD FONT.

Target language	Adaptation data	Preference score				p (t-test)
		NA	W	W+T	N/P	
US English	8 speakers	36.6	41.3	–	22.1	0.109
	×	37.6	–	**47.8**	14.6	0.003
	10% of utts.	–	36.4	**45.1**	18.5	0.003
	8 speakers	34.7	38.2	–	27.1	0.179
	×	29.6	–	**55.6**	14.8	<0.001
	50% of utts.	–	32.8	**51.8**	15.4	<0.001
German	8 speakers	**37.9**	28.5	–	33.7	0.005
	×	26.1	–	**53.3**	20.6	<0.001
	10% of utts.	–	26.4	**51.2**	22.3	<0.001
	8 speakers	31.6	31.6	–	36.8	0.500
	×	20.7	–	**70.8**	8.5	<0.001
	50% of utts.	–	18.3	**73.6**	8.1	<0.001

This is a known issue of all statistical parametric speech synthesizers [31]. Further research to improve the speaker similarity of synthesized speech is required. The MOS test results shows a similar tendency to the preference test for US English conducted in the previous section; the multiple language approaches achieved significantly better scores than the single language approaches.

E. Adaptation to new languages

The third experiment evaluated language adaptation. It was performed as follows:

1) Five SLF models were trained. Each of them was estimated without using one of the five languages in the database.
2) For each SLF model, a language transform was estimated using the data consisting of eight speakers in the excluded language.
3) Then speaker transforms were estimated using all adaptation data from the target speakers.

Note that the target speakers were not included in the data for language adaptation.

A set of paired-comparison preference listening tests were conducted to evaluate the language adaptation process. These tests compared synthesized speech from the SLF model without language adaptation (NA),[20] with the estimated CAT interpolation weights (W), and with the estimated CAT interpolation weights and the additional decision trees (W+T), over 100 (2 speakers × 50 sentences) evaluation utterances. One subject could evaluate a maximum of 40 pairs. Each pair was evaluated by four subjects. Table IV shows the preference test results. It can be seen from the table that having additional decision trees was effective for language adaptation, especially when the target language was far from the training languages. When US English was used for the target language and the remaining four languages (German, UK English, Spanish, and French) were used for training, the estimated points of US English in the language eigenspace were very close to those of UK English. On the other hand, when German was used as the target language, the CAT interpolation weights were roughly evenly distributed. These results indicate that if there exists a language similar to a target language in the training data, reasonable points in the language eigenspace can be found to express the target language. However, if it doesn't exist, estimating CAT interpolation weights is insufficient, thus having additional trees is essential.

[20]The CAT interpolation weights estimated from all training data were used.

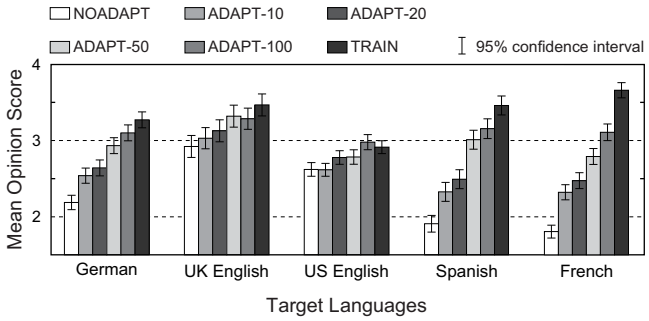

Fig. 7. MOS test results of synthesized speech from the language adapted SLF models using different amounts of adaptation data.

To see the relationship between the performance of language adaptation and the amount of the adaptation data, MOS tests were conducted. The amount of the data for language adaptation was changed (randomly selected 10, 20, 50 to 100 % of adaptation utterances per speaker). The speech samples to be evaluated were synthesized from the language-adapted SLF model with different amounts of adaptation data (ADAPT-X, where X denotes the amount of adaptation data per speaker), the SLF model without language adaptation (NOADAPT),[21] and the SLF model trained with all languages[22] (TRAIN). There were 600 (2 speakers × 50 sentences × 6 systems) samples in the test. One subject could evaluate a maximum of 140 test samples. Each test sample was evaluated by five subjects.

Figure 7 shows the experimental results. It can be seen from the figure that applying language adaptation improved the naturalness of synthetic speech as the amount of adaptation data increased. We can also see that ADAPT-100 achieved similar performance to TRAIN. These results indicate that the SLF framework can adapt to a target condition even when it has a very different context dependency. It also suggests that full SLF training can be well-approximated by incrementally adding a new condition (language) to an existing system.

VI. CONCLUSION

This article has proposed a framework of speaker and language factorization and its application to statistical parametric speech synthesis. This framework factorizes speaker-specific/language-specific characteristics in the data and models them by individual factor-specific transforms. Language-specific factors in the data are represented by transforms based on CAT with cluster-dependent decision trees. Acoustic variations caused by speaker characteristics are handled by transforms based on CMLLR. This form of factorization

[21]The CAT interpolation weights estimated from all training data were used.
[22]This SLF model was the same as the one used in Section V-C.

enables the following things to be done: increasing the quantity of data by having data from multiple speakers in different languages, polyglot speech synthesis, and adding a new language to an existing system.

Future work includes having large variations of languages in the training data. This may remove the requirement for additional decision trees for language adaptation as there is a better chance to find a training language which is similar to a new language. It may enable very rapid language adaptation to be performed. Application of the proposed framework to other factors which have cluster-specific context dependency is also of interest, such as speaking styles, domains, and emotions.

VII. Acknowledgment

The authors would like to thank Dr. Art Blokland for help with data preparation.

References

[1] International Phonetic Association, *Handbook of the international phonetic association*, Cambridge University Press, 1999.
[2] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker adaptive training," in *Proc. ICSLP*, 1996, pp. 1137–1140.
[3] S. Buchholz and J. Latorre, "Crowdsourcing preference tests, and how to detect cheating," in *Proc. Interspeech*, 2011, (to appear).
[4] M. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Comput. Speech Lang.*, vol. 12, no. 2, pp. 75–98, 1998.
[5] ——, "Cluster adaptive training of hidden Markov models," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 417–428, 2000.
[6] ——, "Acoustic factorisation," in *Proc. ASRU*, 2001, pp. 77–80.
[7] R. Kuhn, J. Junqua, P. Nguyen, and N. Niedzielski, "Rapid speaker adaptation in eigenvoice space," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 695–707, 2000.
[8] J. Latorre, K. Iwano, and S. Furui, "New approach to the polyglot speech generation by means of an HMM-based speaker adaptable synthesizer," *Speech Commun.*, vol. 48, no. 10, pp. 1227–1242, 2006.
[9] H. Liang and J. Dines, "An analysis of language mismatch in HMM state mapping-based cross-lingual speaker adaptation," in *Proc. Interspeech*, 2010, pp. 622–625.
[10] Y. Nankaku, K. Nakamura, H. Zen, and K. Tokuda, "Acoustic modeling with contextual additive structure for HMM-based speech recognition," in *Proc. ICASSP*, 2008, pp. 4469–4472.
[11] J. Odell, "The use of context in large vocabulary speech recognition," Ph.D. dissertation, Cambridge University, 1995.
[12] X. Peng, K. Oura, Y. Nankaku, and K. Tokuda, "Cross-lingual speaker adaptation for HMM-based speech synthesis considering differences between language-dependent average voices," in *Proc. ICSP*, 2010, pp. 605–608.
[13] Y. Qian, H. Liang, and F. Soong, "A cross-language state sharing and mapping approach to bilingual (Mandarin-English) TTS," *IEEE Trans. Audio Speech Lang. Process.*, vol. 17, no. 6, pp. 1231–1239, 2009.
[14] K. Saino, "A clustering technique for factor analyzed voice models," Master thesis, Nagoya Institute of Technology, 2008, (in Japanese).
[15] O. Schenk and K. Gärtner, "Solving unsymmetric sparse systems of linear equations with PARDISO," *Journal of Future Generation Computer Systems*, vol. 20, no. 3, pp. 475–487, 2004.
[16] T. Schultz, "Globalphone: a multilingual speech and text database developed at Karlsruhe University," in *Proc. ICSLP*, 2002, pp. 345–348.
[17] K. Shichiri, A. Sawabe, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Eigenvoices for HMM-based speech synthesis," in *Proc. ICSLP*, 2002, pp. 1269–1272.
[18] K. Shinoda and T. Watanabe, "Acoustic modeling based on the MDL criterion for speech recognition," in *Proc. Eurospeech*, 1997, pp. 99–102.
[19] T. Shinozaki, "HMM state clustering based on efficient cross-validation," in *Proc. ICASSP*, 2006, pp. 1157–1160.
[20] K. Sim and M. Gales, "Adaptation of precision matrix models on large vocabulary continuous speech recognition," in *Proc. ICASSP*, 2005, pp. 97–100.
[21] R. Sproat, Ed., *Multilingual text-to-speech synthesis: The Bell labs approach*. Kluwer Academic Publisher, 1998.
[22] T. Toda and K. Tokuda, "A speech parameter generation algorithm considering global variance for HMM-based speech synthesis," *IEICE Trans. Inf. Syst.*, vol. E90-D, no. 5, pp. 816–824, 2007.
[23] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Multi-space probability distribution HMM," *IEICE Trans. Inf. Syst.*, vol. E85-D, no. 3, pp. 455–464, 2002.
[24] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, 2000, pp. 1315–1318.
[25] K. Tokuda, H. Zen, and A. Black, "An HMM-based speech synthesis system applied to English," in *Proc. IEEE Speech Synthesis Workshop*, 2002, CD-ROM Proceeding.
[26] C. Traber, K. Huber, K. Nedir, B. Pfister, E. Keller, and B. Zellner, "From multilingual to polyglot speech synthesis," in *Proc. Eurospeech*, 1999, pp. 835–838.
[27] Y. Wang and M. Gales, "Speaker and noise factorisation on AURORA4 task," in *Proc. ICASSP*, 2011, pp. 4584–4587.
[28] M. Wester, "The EMIME bilingual database," University of Edinburgh, Tech. Rep., 2010.
[29] Y. Wu, Y. Nankaku, and K. Tokuda, "State mapping based method for cross-lingual speaker adaptation in HMM-based speech synthesis," in *Proc. Interspeech*, 2009, pp. 528–531.
[30] J. Yamagishi, "Average-voice-based speech synthesis," Ph.D. dissertation, Tokyo Institute of Technology, 2006.
[31] J. Yamagishi and S. King, "Simple methods for improving speaker-similarity of HMM-based speech synthesis," in *Proc. ICASSP*, 2010, pp. 4610–4613.
[32] J. Yamagishi and O. Watts, "The CSTR/EMIME HTS system for Blizzard Challenge," in *Proc. Blizzard Challenge Workshop*, 2010.
[33] J. Yamagishi, H. Zen, Y. Wu, T. Toda, and K. Tokuda, "The HTS2007' system: Yet another evaluation of the speaker-adaptive HMM-based speech synthesis system in the 2008 Blizzard Challenge," in *Proc. Blizzard Challenge Workshop*, 2008.
[34] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," in *Proc. Eurospeech*, 1999, pp. 2347–2350.
[35] K. Yu and M. Gales, "Adaptive training using structured transforms," in *Proc. ICASSP*, 2004, pp. 317–320.
[36] K. Yu, H. Zen, F. Mairesse, and S. Young, "Context adaptive training with factorized decision trees for HMM-based statistical parametric speech synthesis," *Speech Commun.*, vol. 53, no. 6, pp. 914–923, 2011.
[37] H. Zen, "Speaker and language adaptive training for HMM-based polyglot speech synthesis," in *Interspeech*, 2010, pp. 410–413.
[38] H. Zen and N. Braunschweiler, "Context-dependent additive $\log F_0$ model for HMM-based speech synthesis," in *Proc. Interspeech*, 2009, pp. 2091–2094.
[39] H. Zen, N. Braunschweiler, S. Buchholz, K. Knill, S. Krstulović, and J. Latorre, "HMM-based polyglot speech synthesis by speaker and language adaptive training," in *ISCA SSW7*, 2010, pp. 186–191.
[40] H. Zen, K. Tokuda, and A. Black, "Statistical parametric speech synthesis," *Speech Commun.*, vol. 51, no. 11, pp. 1039–1064, 2009.