Factored Maximum Likelihood Kernelized Regression for HMM-based Singing Voice Synthesis

June Sig Sung, Doo Hwa Hong, Hyun Woo Koo, Nam Soo Kim

School of Electrical Engineering and INMC,
Seoul National University, Korea
{jssung, dhhong, hwkoo}@hi.snu.ac.kr, nkim@snu.ac.kr

Abstract

In our previous work, we proposed factored maximum likelihood linear regression (FMLLR) adaptation where each MLLR parameter is defined as a function of a control vector. In this paper, we introduce a novel technique called factored maximum likelihood kernelized regression (FMLKR) for HMM-based style adaptive speech synthesis. In FMLKR, nonlinear regression between the mean vector of the base model and the corresponding mean vectors of the adaptation data is performed with the use of kernel method based on the FMLLR framework. In a series of experiments on artificial generation of singing voice, the proposed technique shows better performance than the other conventional methods.

Index Terms: style adaptation, singing voice synthesis, MLLR, factored MLLR, kernel method, FMLKR

1. Introduction

Maximum likelihood linear regression (MLLR) is one of the most popular techniques for parameter adaptation in hidden Markov model (HMM)-based systems [1][2]. In the MLLR approach, original parameters of the HMM-based system are mapped to their adapted values via a set of affine transformations which are estimated from a small amount of adaptation data. MLLR was first proposed for speaker adaptation in order to improve the performance of the speech recognition systems, and later a variety of extensions have been developed with applications to other areas [3]-[7].

Generally in MLLR adaptation, the regression parameters are shared among a group of speech units in order to achieve robust parameter estimation with limited amount of data. However, it becomes practically impossible for the MLLR technique to be applied to parameter adaptation when we need separate regression parameters for a huge number of speaking conditions. For instance in singing voice synthesis, it is known that the vocal tract configuration varies depending not only on the phonetic information but also on the musical notes which provide the information concerned with tone and rhythm [8][9].

In our previous work, we extended the conventional MLLR to the factored MLLR (FMLLR) framework where each MLLR parameter is defined as a function of the control parameter vector [10]-[12]. It is similar to cluster adaptive training (CAT) [13] and multiple-regression HMM (MRHMM) [14][15] in using adaptive control of the spectral distributions based on external control parameters. In FMLLR, contrary to them, each element of the MLLR parameter is given as an inner product between a regression vector and a transformed control vector. The control vector contains supplementary information such as the musical notes in singing voice and intensity of each emotion in expressive speech synthesis.

In this paper, we introduce a novel technique called factored maximum likelihood kernelized regression (FMLKR) for HMM-based style-adaptive speech synthesis and compare its performance with MLLR and FMLLR. Recently in the area of speech recognition, Mak et al. proposed the maximum penalized likelihood kernel regression (MPLKR) algorithm for fast speaker adaptation [16] [17]. In MPLKR, kernels were employed in the MLLR framework as the weights of regression vectors, and a penalization term was appended to the likelihood formulation in order to avoid overfitting. The basic idea of this technique is to map the mean vector of the base model to a high-dimensional feature space via a nonlinear mapping before performing linear regression. Similarly to MPLKR, the FMLKR technique performs a nonlinear regression between the mean vector of the base model and the corresponding mean vectors of the adaptation data with the use of kernel methods.

2. FMLLR adaptation

In conventional MLLR adaptation, a p-dimensional mean vector $\boldsymbol{\mu}_s \in R^p$ of a particular distribution s of the HMM is transformed to $\hat{\boldsymbol{\mu}}_s$ via

$$\hat{\boldsymbol{\mu}}_s \quad = \quad \mathbf{M}\,\boldsymbol{\nu}_s \tag{1}$$

where $\mathbf{M}$ is a $p \times (p+1)$ regression matrix which can be decomposed into $\mathbf{M} = [\mathbf{A} \quad \mathbf{b}]$ with $\mathbf{A}$ and $\mathbf{b}$ indicating the parameters of the affine transformation, and $\boldsymbol{\nu}_s$ denotes a $(p+1)$-dimensional augmented mean vector of a particular distribution s defined by

$$\boldsymbol{\nu}_s = [\,\boldsymbol{\mu}_s^{'} \quad 1]^{'} \tag{2}$$

with the prime denoting the transpose of a matrix or a vector [18].

Suppose that for a particular purpose $\mathbf{M}$ should depend on a control parameter $\boldsymbol{\eta}$ which is generally a continuous-valued vector of dimension D. This implies that the mean vector of the distribution s is adapted differently depending on $\boldsymbol{\eta}$. Under this framework, (1) is rewritten as

$$\hat{\boldsymbol{\mu}}_s \quad = \quad \mathbf{M}\,(\,\boldsymbol{\eta})\;\boldsymbol{\nu}_s \tag{3}$$

and an efficient way to achieve (3) is the FMLLR technique [10]-[12].

Let $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T)$ be the given adaptation data vectors. Different from conventional MLLR adaptation, now each adaptation vector $\mathbf{x}_t$ is accompanied with the corresponding $\boldsymbol{\eta}_t$

which denotes the control parameter $\boldsymbol{\eta}$ at time t. The crucial part of the FMLLR approach is to represent $\mathbf{M}(\boldsymbol{\eta})$ as follows:

$$M_{ij}(\boldsymbol{\eta}) = \mathbf{w}_{ij}^{'}\,\boldsymbol{\xi}, \quad 1 \le i \le p, \quad 1 \le j \le p+1 \tag{4}$$

where $M_{ij}(\boldsymbol{\eta})$ indicates the (i,j)-th element of $\mathbf{M}(\boldsymbol{\eta})$ and $\boldsymbol{\xi} = \phi(\boldsymbol{\eta})$ is an L-dimensional control vector obtained by transforming the control parameter $\boldsymbol{\eta}$. Let

$$\mathbf{W} = \{\mathbf{w}_{11}, \mathbf{w}_{12}, \cdots, \mathbf{w}_{1(p+1)}, \mathbf{w}_{21}, \mathbf{w}_{22}, \cdots, \mathbf{w}_{2(p+1)}, \\ \cdots, \mathbf{w}_{p1}, \mathbf{w}_{p2}, \cdots, \mathbf{w}_{p(p+1)}\}$$

denote the set consisting of the L-dimensional regression vectors which are the core parameters of FMLLR.

In order to estimate $\mathbf{W} = \{\mathbf{w}_{ij}\}$, we follow the EM algorithm employed in the conventional MLLR technique. After the posterior probability $\gamma_t(s)$ is computed at the *E* step, we update the parameter $\mathbf{W}$ according to

$$\widehat{\mathbf{W}} = \arg\max_{\mathbf{W}} \mathcal{L}(\mathbf{W}) \tag{5}$$

where

$$\mathcal{L}(\mathbf{W}) = -\frac{1}{2}\sum_{t=1}^{T}\gamma_t(s)\left(\mathbf{x}_t - \mathbf{M}(\boldsymbol{\eta}_t)\,\boldsymbol{\nu}_s\right)^{'}\Sigma_s^{-1} \\ \times\left(\mathbf{x}_t - \mathbf{M}(\boldsymbol{\eta}_t)\,\boldsymbol{\nu}_s\right)$$

in which $\widehat{\mathbf{W}}$ is the updated parameters for FMLLR. Let $\mathbf{W}_i = [\mathbf{w}_{i1}^{'}\ \mathbf{w}_{i2}^{'} \cdots \mathbf{w}_{i(p+1)}^{'}]^{'}$ denote a $(p+1)L$-dimensional vector which concatenates the components of $\mathbf{W}$ corresponding to the i-th element, then the solution to (5) is obtained by setting the gradients of $\mathcal{L}(\mathbf{W})$ with respect to $\mathbf{W}$ to zero as follows:

$$\widehat{\mathbf{W}}_i = \left(\boldsymbol{\nu}_s\,\boldsymbol{\nu}_s^{'} \bigotimes \mathbf{G}_i\right)^{-1}\left(\boldsymbol{\nu}_s \bigotimes \mathbf{r}_i\right) \tag{6}$$

where

$$\mathbf{G}_i = \sum_{t=1}^{T}\gamma_t(s)\frac{1}{\sigma_{s,i}^2}\,\boldsymbol{\xi}_t\,\boldsymbol{\xi}_t^{'}, \tag{7}$$

$$\mathbf{r}_i = \sum_{t=1}^{T}\gamma_t(s)\frac{x_{t,i}}{\sigma_{s,i}^2}\,\boldsymbol{\xi}_t \tag{8}$$

and $\bigotimes$ denotes Kronecker product. Particularly in the case of diagonally structured regression matrix, the parameters are obtained by solving the following equation:

$$\widehat{\mathbf{w}}_{i,i} = \left(\boldsymbol{\nu}_s(i)\,\boldsymbol{\nu}_s(i)^{'} \bigotimes \mathbf{G}_i\right)^{-1}\boldsymbol{\nu}_s(i)\mathbf{r}_i, \quad 1 \le i \le p. \tag{9}$$

For more details, the reader is referred to [10]-[12].

3. FMLKR adaptation

The adaptation scheme given by (3) can be rewritten as follows:

$$\widehat{\boldsymbol{\mu}}_s = \sum_{j=1}^{p+1}\mathbf{M}_j(\boldsymbol{\eta})\,\nu_{s,j} \tag{10}$$

where $\mathbf{M}_j(\boldsymbol{\eta})$ denotes the j-th column vector of the matrix $\mathbf{M}(\boldsymbol{\eta})$ and $\nu_{s,j}$ indicates the j-th element of the augmented mean vector $\boldsymbol{\nu}_s$. From (10), we can see that $\widehat{\boldsymbol{\mu}}_s$ is given by a linear combination of a number of basis vectors. In this interpretation, $\mathbf{M}_j(\boldsymbol{\eta})$ acts as a basis vector and $\nu_{s,j}$ is treated as a weight of the j-th basis vector.

Motivated by this viewpoint, we can extend (10) to a more generalized form as given by

$$\widehat{\boldsymbol{\mu}}_s = \sum_{j=1}^{P}\mathbf{M}_j(\boldsymbol{\eta})\,\psi_j(\boldsymbol{\mu}_s, \boldsymbol{\eta}) \tag{11}$$

where $\{\mathbf{M}_j(\boldsymbol{\eta}), j = 1, 2, \cdots, P\}$ represents a set of basis vectors and $\psi_j(\boldsymbol{\mu}_s, \boldsymbol{\eta})$ is the weight associated with the j-th basis vector. In (11), P denotes the number of basis vectors, which can be selected freely. If $P > p+1$, (11) implies an overcomplete representation. Another point to note in (11) is that $\psi_j(\cdot,\cdot)$ is a nonlinear function which extracts the weight from both the base model parameter $\boldsymbol{\mu}_s$ and the control parameter $\boldsymbol{\eta}$.

A promising way to define the nonlinear function $\psi_j(\cdot,\cdot)$ is to apply a kernel map. Let $\{\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_P\}$ denote a set of P vectors of dimension p and $\{\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \cdots, \boldsymbol{\zeta}_P\}$ be a set of P vectors of dimension D. Then a kernel map is defined by

$$\psi_j(\boldsymbol{\mu}_s, \boldsymbol{\eta}) = \kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right) \tag{12}$$

where $\kappa(\cdot,\cdot)$ denotes a kernel function. Combining (11) and (12), the FMLKR approach adapts the model parameter in the following way:

$$\widehat{\boldsymbol{\mu}}_s = \sum_{j=1}^{P}\mathbf{M}_j(\boldsymbol{\eta})\,\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right). \tag{13}$$

An important issue in FMLKR is how to choose the kernel function $\kappa(\cdot,\cdot)$. In general a kernel function $\kappa(\mathbf{a},\mathbf{b})$ computes how close the two arguments $\mathbf{a}$ and $\mathbf{b}$ are. Since each argument of the kernel function defined in (12) consists of the base model parameter $\boldsymbol{\mu}_s$ and the control parameter $\boldsymbol{\eta}$, a natural way to define the kernel function is to combine two separate kernels defined over respective vectors. In this paper, we apply the kernel function defined by

$$\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right) = \kappa_1(\boldsymbol{\mu}_s, \mathbf{c}_j) + \rho\kappa_2\left(\phi(\boldsymbol{\eta}), \phi(\boldsymbol{\zeta}_j)\right) \tag{14}$$

where

$$\kappa_i(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{\sigma_i^2}\right), \quad i = 1, 2, \tag{15}$$

ρ indicates a ratio between the two Gaussian kernels and $\phi(\cdot)$ is a transformation used for control parameter as in FMLLR.

Similarly to FMLLR, we follow the EM algorithm in order to estimate $\mathbf{W}$ in FMLKR. After the E step, we update the parameter $\mathbf{W}$ according to (5) where the objective function now becomes

$$\mathcal{L}(\mathbf{W}) = -\frac{1}{2}\sum_{t=1}^{T}\gamma_t(s) \\ \times\left(\mathbf{x}_t - \sum_{j=1}^{P}\mathbf{M}_j(\boldsymbol{\eta}_t)\,\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right)\right)^{'} \\ \times\Sigma_s^{-1}\left(\mathbf{x}_t - \sum_{j=1}^{P}\mathbf{M}_j(\boldsymbol{\eta}_t)\,\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right)\right).$$

Note again that each adaptation vector $\mathbf{x}_t$ is accompanied with the corresponding control parameter $\boldsymbol{\eta}_t$. Generally, nonlinear regression with a large number of parameters may usually suffer from the problem of overfitting. In order to alleviate this problem, a regularization technique is usually applied. Based on a regularization strategy, $\mathcal{L}(\mathbf{W})$ is modified to

$$
\begin{aligned}
\mathcal{L}(\mathbf{W}) &= -\frac{1}{2}\sum_{t=1}^{T}\gamma_t(s) \\
&\times \left(\mathbf{x}_t - \sum_{j=1}^{P}\mathbf{M}_j(\boldsymbol{\eta}_t)\,\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right)\right)' \\
&\times \Sigma_s^{-1}\left(\mathbf{x}_t - \sum_{j=1}^{P}\mathbf{M}_j(\boldsymbol{\eta}_t)\,\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right)\right) \\
&- \frac{\beta}{2}\sum_{j=1}^{P}\mathbf{M}_j(\boldsymbol{\eta}_t)'\,\mathbf{M}_j(\boldsymbol{\eta}_t) \qquad (16)
\end{aligned}
$$

where β is a regularization parameter. Applying (14) and the diagonal assumption of the covariance matrix to (16), the parameters are estimated according to the following criterion:

$$
\begin{aligned}
\widehat{\mathbf{W}} &= \arg\max_{\mathbf{W}} -\frac{1}{2}\sum_{t=1}^{T}\gamma_t(s)\sum_{i=1}^{p}\frac{1}{\sigma_{s,i}^2} \\
&\times \left(x_{t,i} - \sum_{j=1}^{P}\mathbf{w}_{ij}'\,\boldsymbol{\xi}_t \cdot \kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right)\right)^2 \\
&- \frac{\beta}{2}\sum_{i=1}^{p}\sum_{j=1}^{P}\left(\mathbf{w}_{ij}'\,\boldsymbol{\xi}_t\right)^2. \qquad (17)
\end{aligned}
$$

The solution to (17) is obtained by setting the derivative of the objective function with respect to $\mathbf{w}_{ij}$ to zero for each row i. Finally, we are led to

$$
\begin{bmatrix}
\mathbf{F}_{i11} + \beta\,\boldsymbol{\xi}_t\,\boldsymbol{\xi}_t' & \mathbf{F}_{i12} & \cdots & \mathbf{F}_{i1P} \\
\mathbf{F}_{i21} & \mathbf{F}_{i22} + \beta\,\boldsymbol{\xi}_t\boldsymbol{\xi}_t' & \cdots & \mathbf{F}_{i2P} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{F}_{iP1} & \mathbf{F}_{iP2} & \cdots & \mathbf{F}_{iPP} + \beta\,\boldsymbol{\xi}_t\,\boldsymbol{\xi}_t'
\end{bmatrix}
\times
\begin{bmatrix}
\hat{\mathbf{w}}_{i1} \\ \hat{\mathbf{w}}_{i2} \\ \vdots \\ \hat{\mathbf{w}}_{iP}
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{G}_{i1} \\ \mathbf{G}_{i2} \\ \vdots \\ \mathbf{G}_{iP}
\end{bmatrix}
$$

where

$$
\begin{aligned}
\mathbf{F}_{ijk} &= \sum_{t=1}^{T}\frac{\gamma_s(t)}{\sigma_{s,i}^2}\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right) \\
&\quad \cdot \kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_k, \boldsymbol{\zeta}_k)\right)\,\boldsymbol{\xi}_t\,\boldsymbol{\xi}_t', \qquad (18)
\end{aligned}
$$

$$
\mathbf{G}_{ij} = \sum_{t=1}^{T}\frac{\gamma_s(t)}{\sigma_{s,i}^2}x_{t,i}\kappa\left((\boldsymbol{\mu}_s, \boldsymbol{\eta}_t), (\mathbf{c}_j, \boldsymbol{\zeta}_j)\right)\,\boldsymbol{\xi}_t'. \qquad (19)
$$

4. Experiments on singing voice synthesis

The objective in this experiment is to apply the transform methods presented in Sections 2-3 to adapt the HMM parameters of a reading-style speech synthesizer to a set of given singing voice. Since it is difficult to collect a large amount of singing voice and reading-style speech simultaneously from the same speaker, we attempted to adapt the parameters of a reading-style speech synthesizer with a small amount of singing voice data. For the construction of reading-style speech synthesizers, we used the Korean speech data spoken by two female speakers: YMK and SJK. The speaker YMK provided only the reading-style speech data while both the reading-style and singing voice data were available for the speaker SJK. The reading-style speech synthesizer for the speaker YMK was trained with 4,000 utterances amounting to 525 minutes. On the other hand, the reading-style speech synthesizer for the speaker SJK was obtained by adapting the parameters of the speaker YMK with 162 utterances amounting to 32 minutes.

Each utterance was sampled at 16 kHz and a 20 ms Hamming window was applied with 5 ms frame shift for speech feature extraction. As for the spectrum feature, a 25th-order mel-scaled cepstrum vector was extracted at each frame. By attaching the Δ- and $\Delta\Delta$-cepstra derived from the extracted mel-scaled cepstrum sequence, the spectrum feature could be represented by a 75-dimensional vector at each frame. We also extracted the pitch from each frame for the generation of voiced excitation signals. As the basic unit of speech synthesis, we applied quinphones followed by context-dependent reading-style text analysis described in [19]. Each quinphone was modeled by a 5 state left-to-right structured HMM where the observation distribution at each state was given by a single Gaussian PDF with diagonal covariance matrix.

The parameters of the HMM-based speech synthesizer for the reading-style speech were trained by following the general technique presented in [20]. For a robust parameter estimation, the decision tree technique was employed to share the observation distributions across the states, which resulted in 3,198 leaf nodes for the spectrum of the speaker YMK. The parameters of the reading-style speech synthesizer for the speaker YMK were adapted to the utterances of the speaker SJK based on the speaker adaptation method supported by HTS [20]. The number of transform matrices in this procedure was 819. The adapted parameters then constructed the reading-style speech synthesizer for the speaker SJK.

To build a singing voice database, we collected 95 Korean songs amounting to 105 minutes sung by the speaker SJK. Spectrum features of the singing voice data were extracted and represented in the same manner with those of the reading-style utterances. In conjunction with the recorded sounds, the musical score associated with each song was also provided.

The method of signal generation in the singing voice synthesizer is almost the same to that of the reading-style speech synthesizer except that the lyrics are synchronized with the musical notes which control the pitch and duration. When synthesizing the singing voice, we applied the values of the note such as the pitch and duration to the HMM directly without any statistical modeling which means that the same pitch and duration were applied to generate the synthesized singing voice for each musical note and corresponding syllable.

4.1. Objective performance evaluation for singing voice synthesis

Five different methods were compared in the experiments. We tried two kinds of MLLR with different structures of the regression matrix $\mathbf{M}$: diagonal (MLLR_d) where the first p columns of $\mathbf{M}$ constitute a diagonal matrix and the unrestricted full matrix (MLLR_f). The third and fourth methods are denoted by

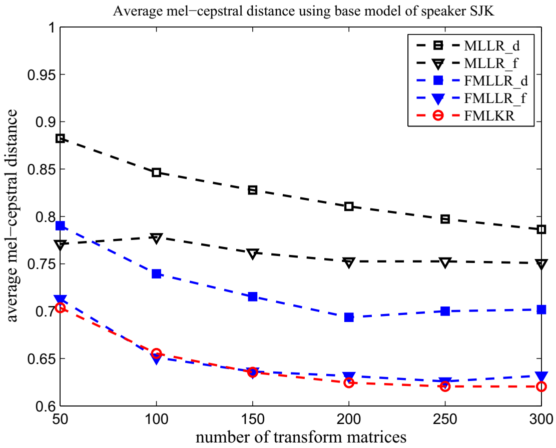

Figure 1: Average mel-cepstral distance between the original and synthesized singing voice generated by speech synthesizer based on speaker SJK.

FMLLR_d and FMLLR_f in which the HMM mean vectors were transformed differently based on the diagonally structured and fully structured regression matrices, respectively. In the fifth method denoted as FMLKR the adapted HMM mean vectors were expressed by the transformation based on the FMLKR technique as given in (18).

Among the 95 songs provided by the speaker SJK, we used 80 songs for training the regression matrices of each method and the other 15 songs for evaluating performances. Considering the size of the decision tree for the reading-style speech synthesizer, the amount of the singing voice data was considered too small to expect a good adaptation performance. To alleviate this difficulty, we applied a tying approach to the estimation of the regression matrices.

When applying FMLLR and FMLKR, the pitch and duration derived from each musical note were used as the control parameter, $\boldsymbol{\eta} = (\tilde{P}, \tilde{D})$ where $\tilde{P}$ is the fundamental frequency of the note written in the unit of Hz and $\tilde{D}$ indicates the duration given as the number of frames. In this paper, we tried a configuration as given by

$$\boldsymbol{\xi}_t = \left(1, \log\tilde{P}(t), \log\tilde{D}(t)\right)', \qquad (20)$$

where $\tilde{P}(t)$ and $\tilde{D}(t)$ indicate $\tilde{P}$ and $\tilde{D}$ at time t, respectively. For FMLKR, we set β, ρ, σ_1^2 and σ_2^2 in (14) to 0.5, 0.01, 1.0 and 0.1, respectively, which were found suitable from a number of preliminary experiments. $\mathbf{c}_j$ and $\boldsymbol{\zeta}_j$ were respectively obtained by clustering the mel-cepstra and control parameters in the singing voice database.

Fig. 1 shows the average mel-cepstral distance between the original and synthesized singing voices obtained from the speaker SJK with varying number of regression matrices. In this evaluation, we set P to 25 for FMLKR with which the number of transformation parameters could be kept the same with those of other methods. Even though the average mel-cepstral distance cannot strictly prove the quality or naturalness of the synthetic speech generated by each method, it is considered a reasonable measure to evaluate how similar or different the generated mel-cepstrum is from the original data. From the results we can find that both FMLLR and FMLKR approaches much reduced the mel-cepstral discrepancy than other approaches.

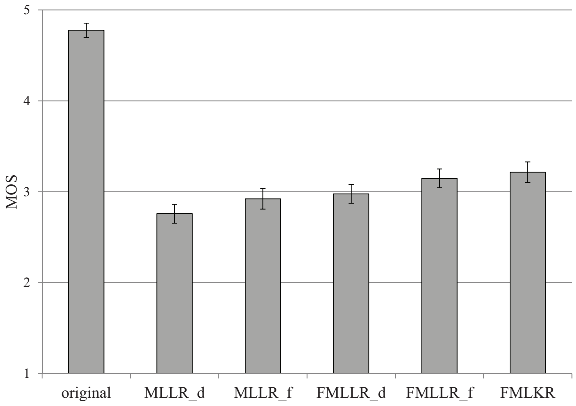

Figure 2: Result of MOS test for singing voice. Note that the lines in the top ends of the bars indicate the 95% confidence intervals.

4.2. Subjective listening test for singing voice synthesis

We performed a subjective listening test in which 14 listeners participated. For the test, the mean opinion score (MOS) test was performed to evaluate the overall quality of the synthesized singing voices generated by various different methods. The MOS is expressed as a single number in the range 1 to 5, where 1 indicates the lowest perceived speech or singing voice quality and 5 the highest [22].

The five methods were applied to generate the singing voice of 8 songs which were not included in the training database. The control vector for the FMLLR and FMLKR methods was determined as (20) and the number of regression matrices for each method was set the same at 250. From the obtained scores shown in Fig. 2, we can see that the proposed approach produced a better quality than the other approaches to singing voice synthesis.

5. Conclusions

In this paper, we have proposed the FMLKR approach as a novel technique for adapting the HMM parameters when the adaptation should depend on varying control parameters. The proposed approach performs a nonlinear regression between the mean vector of the base model and the corresponding mean vectors of the adaptation data with the use of kernel methods. Moreover, we have described MLLR, FMLLR and FMLKR techniques in the common ML framework and compared their performance when applied to adapting the parameters of a reading-style speech synthesizer to the singing voice data. From the experimental results, it has been found that FMLKR outperformed the traditional MLLR techniques in terms of an objective measure such as the mel-cepstral distance as well as the subjective listening quality measure.

6. Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2012R1A2A2A01045874) and the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2012-H0301-12-2005).

7. References

[1] C. J. Leggetter, and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Comput. Speech Lang.*, vol. 9, no. 2, pp. 171-185, Apr. 1995.

[2] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Comput. Speech Lang.*, vol. 12, no. 2, pp. 79-98, 1998.

[3] M. J. F. Gales, "Cluster adaptaive training of hidden Markov Models," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 417-428, Jul. 2000.

[4] K. Visweswariah, V. Goel, and R. Gopinath, "Structured linear transforms for adaptation using training time information," *Proc. ICASSP*, pp. 585-288, 2002.

[5] B. Mak, and R. Hsiao, "Kernel eigenspace-based MLLR adaptation," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 15, no. 3, pp. 784-795, Mar. 2007.

[6] Z. Karam, and W. Campbell, "A multi-class MLLR kernel for SVM speaker recognition," in *Proc. ICASSP*, Las Vegas, NV, pp. 4117-4120, 2008.

[7] Y. Sung, C. Boulis, and D. Jurafsky, "Maximum conditional likelihood linear regression and maximum a posteriori for hidden conditional random fields speaker adaptation," in *Proc. ICASSP*, Las Vegas, NV, pp. 4293-4296, 2008.

[8] J. Sundberg, "The acoustics of the singing voice," *Sci. Amer.*, pp. 82-91, Mar. 1977.

[9] E. Joliveau, J. Smith, and J. Wolfe, "Vocal tract resonances in singing: The soprano voice," *The Journal of the Acoustical Society of America*, Vol. 116, pp. 2434-2439, 2004.

[10] N. S. Kim, J. S. Sung, and D. H. Hong, "Factored MLLR adaptation," *IEEE Signal Processing Letters*, vol. 18, no. 2, pp. 99-102, Feb. 2011.

[11] J. S. Sung, D. H. Hong, S. J. Kang, and N. S. Kim, "Factored MLLR adaptation for singing voice generation," *Proc. Interspeech*, pp. 2789-2792, Firenze, Italy, Aug. 2011.

[12] J. S. Sung, D. H. Hong, H. W. Koo and N. S. Kim, "Factored MLLR Adaptation Algorithm for HMM-based Expressive TTS," *Proc. Interspeech*, Portland, Or., Sep. 2012.

[13] J. Latorre, V. Wan, M. J. F. Gales, L. Chen, K. K. Chin, K. Knill, and M. Akamine, "Speech factorization for HMM-TTS based on cluster adaptive training," in *Proc. Interspeech*, Portland, Or., Sep. 2012.

[14] T. Nose, J. Yamagishi, T. Masuko, and T. Kobayashi, "A style control technique for HMM-based expressive speech synthesis," *IEICE Trans. Inf. Sys.*, vol. E90-D, no. 9, pp. 1406-1413, Sep. 2007.

[15] Z. -H. Ling, K. Richmond, and J. Yamagishi, "Articulatory control of HMM-based parametric speech synthesis using feature-space-switched multiple regression," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 21, no. 1, pp. 207-219, Jan. 2013.

[16] B. Mak, and R. Hsiao, "Kernel eigenspace-based MLLR adaptation," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 15, no. 3, pp. 784-795, Mar. 2007.

[17] B.Mak, T. Lai, I. Tsang, and J. Kwok, "Maximum Penalized Likelihood Kernel Regression for Fast Adaptation", *IEEE Trans. Audio, Speech Lang. Process.*, vol. 17, no. 7, pp. 1372-1381, Sep. 2009.

[18] S. Young, et. al, *The HTK book,* Cambridge University Engineering Department, pp. 136-147, 2006.

[19] J. S. Sung, D. H. Hong, K. H. Oh, and N. S. Kim, "Excitation modeling based on waveform interpolation for HMM-based speech synthesis," in *Proc. Interspeech*, Makuhari, Japan, pp. 813-816, Sep. 2010.

[20] H. Zen et al., "The HMM-based speech synthesis system version 2.0," in *Proc. of ISCA SSW6*, Bonn, Germany, Aug. 2007.

[21] K. Saino et al., "HMM-based singing voice synthesis system," in *Proc. Interspeech*, Pittsburgh, PA, pp. 1141-1144, Sep. 2006.

[22] V. Grancharov, and W. Kleijn, "Speech quality assessment", Springer Handbook of Speech Processing, chap. 5, 2007.