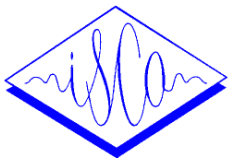

3rd International Conference on
Spoken Language Processing (ICSLP 94)
Yokohama, Japan
September 18-22, 1994

Voice Adaptation Using Multi-Functional Transformation with Weighting by Radial Basis Function Networks

Naoto Iwahashi &

SONY Research Center
Kitashinagawa, Shinagawa-ku, Tokyo, 141, Japan
naoto@av.crl.sony.co.jp

Yoshinori Sagisaka

ATR Interpreting Telecommunications
Research Labs.
Seika-cho Soraku-gun, Kyoto 619-02, Japan
sagisaka@itl.atr.co.jp

ABSTRACT

This paper describes a spectral transformation method for voice conversion, using multiple linear functions with weighting by Radial Basis Function (RBF) networks. Spectral transformation by speaker interpolation with single linear function is the method which can obtain a moderate mapping by using a small amount of training data. However, even if larger amounts of data could be used, a more precise mapping can not be obtained. To cope with this, multiple linear functions with weighting are used. The weight value is decided by a weighting function represented by RBF networks. Parameters of both the linear functions and the weighting function are simultaneously adapted. The reduction rate of the spectral distance from the generated spectrum to the target speaker, compared with the distance from the interpolated speaker closest to the target, was calculated. It was shown that while the distance reduction rate was about 42 % using the single linear function, the rate increased to 48 % using the multi-functional transformation, which includes two linear functions.

1. INTRODUCTION

In a spectral transformation method for voice conversion, it is desired that adaptation for a target speaker is suitably carried out according to the amount of training data from the target speaker. Previously, several adaptation methods have been proposed. A method based on VQ-codebook mappings[1] needs a very large amount of training data for target speaker utterances in order to provide suitably corresponding VQ-codes. This is due to the fact that suitable constraints are not used, that is, each VQ-code correspondence is independently considered. On the other hand, a method based on interpolating pre-stored multiple speakers' utterance data to generate new spectrum patterns[2] can provide comparatively good adaptation when using small amounts of training data, *e.g.* one word utterance. However, in this speaker interpolation with the single linear function, while moderate mapping into the target speaker's spectrum can be obtained, a more precise mapping cannot be obtained even if larger amounts of training data could be used. In order to get a more precise mapping using large amounts of training data, more freedom for the adaptation is needed.

To cope with this problem, we propose a multi-functional transformation method which uses multiple linear functions with weighting. The weight value for each linear function is decided by a weighting function represented by Radial Basis Function networks[3]. Parameters of this multi-functional transformation are adapted by the gradient descent method, where both the linear functions and the weighting function are simultaneously adapted to training data. This multi-functional transformation is regarded as a general technique which provides minimum non-linearity which is necessary.

In the following sections, after briefly describing the previously proposed speaker interpolation with single linear function, multi-functional transformation will be described.

2. SPEAKER INTERPOLATION

In spectrum transformation by speaker interpolation, the spectrum data of utterances spoken by multiple speakers is pre-stored and employed through interpolation. After the spectrum sequences of the same utterance by multiple speakers are time-aligned by DTW, the interpolation is carried out between these DTWed spectrum sequences by the following transformation:

$$\begin{aligned} \mathbf{Y}_i &= F_a(\mathbf{X}_i) \\ &= \mathbf{X}_i \cdot \mathbf{a} + \mathbf{b} \end{aligned} \quad (1)$$

where

$$\mathbf{X_i} = \begin{bmatrix} x_{i11} & \cdot & \cdot & x_{i1K} \\ \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot \\ x_{iJ1} & \cdot & \cdot & x_{iJK} \end{bmatrix}$$

$$\mathbf{Y}_i^T = [Y_{i1}, Y_{i2}, ..., Y_{iJ}]$$

$$\mathbf{a}^T = [\, a_1, ..., a_K \,]$$

$$\mathbf{b}^T = [\, b_1, ..., b_J \,]$$

Here, x_{ijk} represents the jth spectral parameter (*e.g.* cepstrum) of the ith frame in a DTWed utterance of the kth speaker. J is the order of the spectral parameters. K is the number of pre-stored speakers. Y_{ij} represents the jth spectral parameter of the ith frame of the spectrum generated by interpolation. a_k is the interpolation coefficient for the kth speaker, and b_j is the jth cepstrum parameter. The parameters in the transformation

in Equation 1 are decided by solving normal equations so as to minimize an averaged squared error between the spectrum of the target speaker and the spectrum generated by this transformation.

3. USE OF MULTIPLE LINEAR FUNCITIONS

Advantage

The single linear function in Equation 1 is not always enough to precisely represent the mapping characteristics from multiple speakers' spectra to those of the target speaker, because localities of mapping characteristics may remain even though speech spectrum modelling by multi speakers' spectra would be effective. Therefore, use of multiple linear functions can be considered, where each linear function is applied to spectral subdivisions of the entire spectral space. This subdivided space scheme can be extended to give weight vectors for each linear function at any point in the spectral space. This can be regarded as a fuzzy subdividing of the space. The formulation of the mapping transformation by weighted multiple linear functions is

$$\begin{aligned} \mathbf{Y}_i &= [\, F_{a1}(\mathbf{X}_i),\ \ldots\ , F_{aL}(\mathbf{X}_i)\,] \cdot \mathbf{g}_i \\ &= (\mathbf{X}_i \cdot \mathbf{A} + \mathbf{B}) \cdot \mathbf{g}_i \end{aligned} \qquad (2)$$

where

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdot & \cdot & a_{1L} \\ \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot \\ a_{K1} & \cdot & \cdot & a_{KL} \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} b_{11} & \cdot & \cdot & b_{1L} \\ \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot \\ b_{J1} & \cdot & \cdot & b_{JL} \end{bmatrix}$$

$$\mathbf{g}_i^T = [\, g_1(\mathbf{X}_i),\ \ldots\ , g_L(\mathbf{X}_i)\,]$$

with a constraint

$$\sum_{l=1}^{L} g_l(\mathbf{X}_i) = 1 \qquad (\forall i\ ). \qquad (3)$$

Here, L is the number of linear functions, $F_{al}(\cdot)$ is the lth linear function, a_{kl} is the weighting value for the kth speaker in the lth function, b_{jl} is an additive constant of jth order in the lth linear function, $g_l(\cdot)$ is a weighting function in which the input is the multi speakers' spectral vector and the output is a weighting value for the lth linear function, and a_{lk} is a weighting value for the kth speaker in the lth linear function. A structure of the multiple linear functional representation with weighting function ($L = 2$) is shown in Figure 1. This is similar to the modular connectionist architecture[4].

The weighting function fuzzy-subdivides the whole spectral space to catch the local characteristics on the spectral transformation. Considering the representation of this locality, Radial Basis Function networks[3] are attractive to use, as they can produce a localized response and are suitable for localizing each linear function's coverage in spectral space.

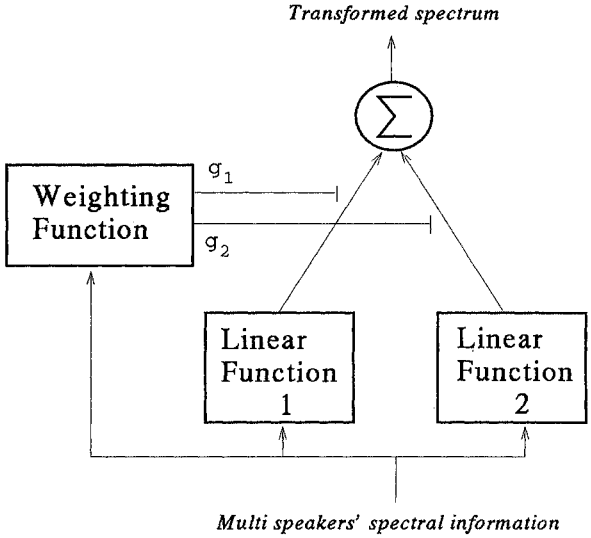

Figure 1. Block diagram of multi-functional representation with weighting function

Weighting by RBF network

A structure of the weighting function using RBF networks is shown in Figure 2. The radial basis functions in the hidden layer are gaussian kernel functions,

$$\begin{aligned} o_q &= \exp\left\{-\frac{\|\mathbf{z} - \mathbf{c}_q\|}{2\sigma_q^2}\right\} \\ &= \exp\left\{-\frac{1}{2\sigma_q^2}\sum_{m=1}^{M}(z_m - c_{qm})^2\right\}. \end{aligned}$$

Here, z_m is the input value to the RBF network where the input dimension is M. $\mathbf{c}_q$ and σ_q are the center vector and the normalizing factor in the qth gaussian kernel function. o_q is the output of the qth gaussian kernel function. To satisfy the constraint for the output of the weighting function in Equation 3, the output node is modified to produce normalized values for multiple input values. The output node works as the function,

$$g_p = \frac{w_p o_p}{\sum_{l=1}^{L} w_l o_l} \qquad (p = 1, \ldots, L)$$

where

$$\sum_{l=1}^{L} w_l = 1.$$

Here, w_l is a link weight. g_p is the pth output value of the weighting function. This output node can be regarded as a modification of the multipartitioning unit[5].

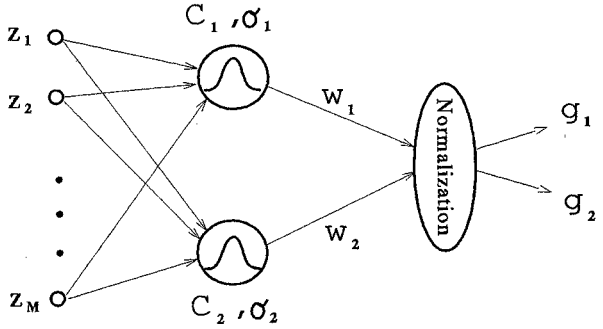

Figure 2. Structure of the weighting function by radial basis function networks with output normalization

Optimization procedure

Parameters of the above-mentioned multi-functional model are optimized to minimize Q, the sum of squared errors between the responses of this model and the values of the target speaker's spectrum parameters in the learning samples. Q can be calculated as follows:

$$\begin{aligned} Q &\equiv \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{Y}_i\| \\ &= \sum_{i=1}^{N} \left\{ \sum_{j=1}^{J} (y_{ij} - \sum_{l=1}^{L} g_{il} (\sum_{k=1}^{K} a_{kl} x_{ijk} + b_{jl}))^2 \right\} \end{aligned}$$

Here, g_{il} is the lth output value of the weighting function for the ith learning sample. N is the number of training samples.

The learning of this multi-functional model can be carried out by breaking the problem down into two processes. One is an optimization process of the linear functions, and the other is an update process of the nonlinear weighting function. These two processes are done alternately in an iterative fashion.

Assuming that weighting values g_{il} $(i = 1, ..., N, l = 1, ..., L)$ are fixed, the values of the parameters a_{kl}, b_{jl} in the linear functions are determined uniquely by solving the following linear equations obtained from the partial derivatives of Q, providing that the coefficient vectors in the left hand side of these equations are linearly independent of each other.

$$\sum_{i=1}^{N} \sum_{j=1}^{J} \left\{ g_{iq} x_{ijp} \sum_{l=1}^{L} g_{il} \left(\sum_{k=1}^{K} x_{ijk} a_{kl} + b_{jl} \right) \right\} = \sum_{i=1}^{N} \sum_{j=1}^{J} y_{ij} g_{iq} x_{ijp}$$

$$\sum_{i=1}^{N} \left\{ g_{is} \sum_{l=1}^{L} g_{il} \left(\sum_{k=1}^{K} x_{irk} a_{kl} + b_{rl} \right) \right\} = \sum_{i=1}^{N} y_{ir} g_{is}$$

$$(p = 1, .., K \quad q = 1, .., L \quad r = 1, .., J \quad s = 1, .., L)$$

Parameters in the weighting function shown in Figure 2 are updated by the gradient descent method. As for the parameter c_{rs}, the sth element of the center vector for the rth gaussian kernel function, it is determined iteratively according to

$$c_{rs}(t+1) = c_{rs}(t) - \mu \left. \frac{\partial Q}{\partial c_{rs}} \right|_{\Phi(t)} \tag{4}$$

where μ is a positive constant, learning rate. $\Phi(t)$ represents the values of all parameters in the multi-functional model in Figure 1 at the tth iteration. The partial derivative of Q with respect to c_{rs} can be written using the Chain Rule

$$\begin{aligned} \frac{\partial Q}{\partial c_{rs}} &= \sum_{i=1}^{N} \frac{\partial d_i}{\partial c_{rs}} \\ &= \sum_{i=1}^{N} \sum_{p=1}^{L} \frac{\partial d_i}{\partial g_{ip}} \frac{\partial g_{ip}}{\partial c_{rs}} \\ &= \sum_{i=1}^{N} \sum_{p=1}^{L} \frac{\partial d_i}{\partial g_{ip}} \frac{\partial g_{ip}}{\partial o_{ir}} \frac{\partial o_{ir}}{\partial c_{rs}} \end{aligned}$$

where

$$\begin{aligned} \frac{\partial d_i}{\partial g_{ip}} &= \sum_{j=1}^{J} \left\{ 2 \left(\sum_{k=1}^{K} x_{ijk} a_{kp} \right) \left(\sum_{k=1}^{K} \sum_{l=1}^{L} g_{il} x_{ijk} a_{kl} \right) \right. \\ &\quad + 2 b_{jp} \sum_{l=1}^{L} (g_{il} b_{jl}) - 2 y_{ij} \sum_{k=1}^{K} (x_{ijk} a_{kp}) \\ &\quad - 2 y_{ij} b_{jp} + 2 b_{jp} \sum_{k=1}^{K} \sum_{l=1}^{L} g_{il} x_{ijk} a_{kl} \\ &\quad \left. + 2 \sum_{k=1}^{K} (x_{ijk} a_{kp}) \sum_{l=1}^{L} (g_{il} b_{jl}) \right\} \\ \frac{\partial g_{ip}}{\partial o_{ir}} &= w_p \frac{1}{\sum_{l=1}^{L} w_l o_{il}} - \frac{w_p^2 o_{ip}}{\left(\sum_{l=1}^{L} w_l o_{il} \right)^2} \qquad (p = r) \\ &= - \frac{w_p w_r o_{ip}}{\left(\sum_{l=1}^{L} w_l o_{il} \right)^2} \qquad (p \neq r) \\ \frac{\partial o_{ir}}{\partial c_{rs}} &= - \frac{1}{\sigma_r^2} (-z_{is} + c_{rs}) \exp \left(- \frac{1}{2\sigma_r^2} \sum_{m=1}^{M} (z_{im} - c_{rm})^2 \right). \end{aligned}$$

Here, z_{im} is the mth input value for the ith training sample. o_{ir} is the output value of the rth gaussian kernel function of the ith training sample.

The other parameters, such as σ_l and w_l, are similary updated.

4. EVALUATION AND DISCUSSION

The adaptation carried out by using the multiple linear functions weighted by RBF networks was tested. The analysis conditions for spectral parameters are shown in Table 1. The number of pre-stored speakers was four - two males and two females. In order to check the basic ability of the multi-functional model, two linear functions were used, and only the center vectors in the weighting function were adapted. The initial center vectors of the gaussian kernel functions were given as vectors which were generated by adding small perturbations to a centroid vector in the input training data. σ_l and

Table 1. Analysis conditions

Sampling rate	12 kHz
Frame period	5 ms
Window width	21.3 ms
Spectral parameter	30 order LPC-cepstrum (12 order LPC)

w_l for $l = 1, 2$ in the weighting function were fixed to 0.5 and 0.5 respectively. The learning rate μ in Equation 4 was set to 0.001. No attempt was made to optimize the choice of this value. One male speaker was used as the target speaker in the experiments. Ten word utterances were used as training data. For comparison, the single linear function represented in Equation 1 was also adapted using the same training data.

In experiments, the reduction rate of the log-spectral distance from the generated spectrum to the target speaker's spectrum, compared with the distance from the spectrum of the interpolated speaker closest to the target, was calculated. The results are shown in Table 2. We can see that whilst the distance reduction rate from using the single linear function (SLF) was 42 %, the rate from the multiple linear functions (MLF) increased to 48 %. This also shows that over-adaptation occured to some extent when using multi-functions, because there were larger differences between the reduction rates at training and testing in the MLF case in comparison with the SLF case. More training data might improve the performance. We should select the number of the linear functions according to the amount of training data by an open data generality check method, such as cross validation. In addition, Figure 3 shows the decreasing of the distance by the iterative adaptation procedure for the multiple functional model described in Section 3. These results confirmed that the proposed non-linear optimization method worked well.

5. CONCLUSION

A new speech spectrum transformation method for voice conversion in speech synthesis, using multiple linear functions with weighting by Radial Basis Function networks, has been described in this paper. In this speaker interpolation scheme, multiple linear functions with weighting were used to get a more precise mapping to a target speaker. The weight value for each linear function was decided by a weighting function represented by RBF networks. Parameters of this multi-functional transformation were trained by using the gradient descent method, where both the linear functions and the weighting function were simultaneously adapted. The speech spectrum could be transformed and adapted to the target speaker's spectrum by the proposed methods. By using ten words as training data, whilst the distance reduction rate in the case of using the single linear function was 42 %, the rate of the multiple linear functions increased to 48 %. We can select the number of linear functions for adaptation according to the amount of training data. The proposed multi-functional transformation is a robustly trainable non-linear transformation against small training data. This transformation would be applicable to other areas, such as feature extraction in speech recognition.

Table 2. Comparison of distance reduction rates (%) between the single linear function case (SLF) and the multiple linear functions case (MLF) with ten learning words

Method	Training	Testing
SLF	44	42
MLF	52	48

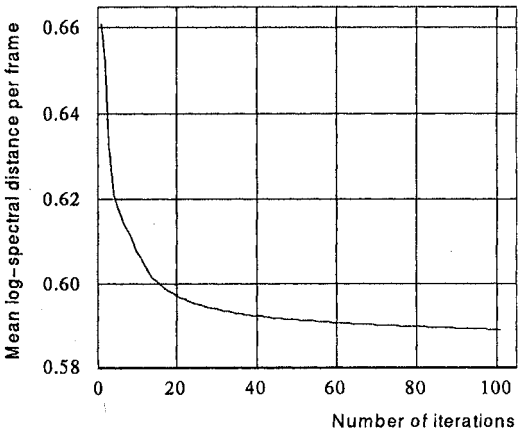

Figure 3. Decrease of log-spectral distance in the non-linear optimization process for the multi-functional model

References

[1] M. Abe, S. Nakamura, K. Shikano and H. Kuwabara, "Voice conversion through vector quantization", *Proc. ICASSP*, pp.655-658, 1988.

[2] N. Iwahashi and Y .Sagisaka, "Speech spectrum transformation by speaker interpolation", *Proc. ICASSP*, Vol.1, pp.461-464, 1993.

[3] D. S. Broomhead and D. Lowe, "Radial Basis Functions, Multi-Variable Function Interpolation and Adaptive Networks", *Royal Signals and Radar Establishment Memorandum 4148*, 1988.

[4] R. A. Jacobs, M. I. Jordan, S. J. Nowlan and G. E. Hinton, "Adaptive mixtures of local experts", *Neural Computation*, Vol.3, No.1, pp.79-87, 1991.

[5] Y. Tan and T. Ejima, "A network with multipartitioning units", *Proc. International Joint Conference on Neural Networks*, Washington, D.C., pp.II 439-442, 1989.