Acoustic Model Training Using Feature Vectors Generated by Manipulating Speech Parameters of Real Speakers

Tetsuto Kawai*, Norihide Kitaoka* and Kazuya Takeda*
* Nagoya University, Aichi, Japan
E-mail: kawai.tetsuto@g.sp.m.is.nagoya-u.ac.jp

***Abstract*—In this paper, we propose a robust speaker-independent acoustic model training method using generative training to generate many pseudo-speakers from a small number of real speakers. We focus on the difference between each speaker's vocal tract length, and manipulate it in order to create many different pseudo-speakers with a range of vocal tract lengths. This method employs frequency warping based on the inverted use Vocal Tract Length Normalization(VTLN). Another method for creating pseudo-speakers is to vary the speaking rate of the speakers. This can be achieved by a method called PICOLA; Pointer Interval Controlled OverLap and Add. In experiments, we train acoustic models using these generated pseudo-speakers in addition to the original speakers. Evaluation results show that generating pseudo-speakers by manipulating speaking rates did not result in a sufficient increase in performance, however, vocal tract length warping was effective.**

I. Introduction

Speech recognition techniques are widely spread in many kinds of applications; car navigation systems, speech guidance, and so on. These application require consistent high performance of speaker-independent speech recognition systems. To the robust recognition, the acoustic models used in the speech recognition system plays a central role. In the training process of acoustic models, using a small training data sets during the training process of acoustic models can lead to poor recognition rates. In particular, recognition performance is insufficient for unknown speakers whose data is not used in the training process. It is important to make use of training data from many speakers to make models robust to even unknown speakers. However, collecting training data from such a lot of people is very difficult. In our study, methods of training robust speaker-independent acoustic models from small number of speakers are proposed.

In order to improve the recognition performance, adaptation techniques are often used. Model-based adaptations, such as maximum a posteriori (MAP) adaptation[1] and maximum likelihood linear regression (MLLR) [2], transform acoustic models (usually hidden Markov models (HMMs)) to fit the target speaker or environment. Particularly in conventional study, constrained MLLR (CMLLR) is often proposed [3]. Previously we proposed a method based on the opposite view [4], in which we do not remove the speaker variations; instead we added them to the averaged speech features. We assumed that individual speech variation is generated by adding individual differences to an "average" person. Speaker recognition using the MLLR transformation matrix [5] suggests that the linear transformation matrix expresses individuality. We first obtained the MLLR transformation matrices for training speakers from a limited amount of speech data and applied PCA to them to extract a small number of bases. Then we generated pseudo-speaker transformation matrices from the statistical linear combination of the bases. Finally, the speech features were generated by applying the inverse transformation matrices to the normalized speech features and we trained the speaker independent (but environment adapted) acoustic models using generated speakers.

When we applied this method in our previous study, however, we did not pay enough attention to speakers' speech characteristics (vocal tract length, speaking rate, etc.). It is not clear whether speakers generated using this method reflected characteristics of real speakers or not.

In this paper, when generating pseudo-speakers, we focus on characteristics such as vocal tract length and speaking rate, and express each speaker's characteristics explicitly. First, we pay attention to the fact that everyone has a different vocal tract length. By compressing and expanding a speaker's vocal tract virtually, we can express many types of voices. This can be achieved by the reverse use of Vocal Tract Length Normalization (VTLN), which is usually used to normalize the vocal tract length of speakers. In our proposed method, we change the speakers' vocal tract lengths at random and make many types of voices, instead of normalizing them.

Another method we use in order to express different characteristics is to change speakers' speaking rates. Compared to manipulating their vocal tract length, which mimics differences in speakers' physical characteristics, this method can be used to express their unique prosodic characteristics. It is important to focus on these characteristics, especially when training models robust to spontaneous speech. We use the PICOLA algorithm, which only changes the speaker's speaking rate without changing the pitch of the voice, or the original voice quality.

In this study, we propose using these two methods to express the speakers' characteristics explicitly. We can then obtain a large number of speech variations from a small number of speakers, and make robust speaker-independent acoustic models.

The rest of this paper consists of four sections. In section II and section III, we illustrate how to virtually change the length of a speaker's vocal tract and speaking rate in detail. In section IV, we train acoustic models using not only existing speakers, but also using pseudo-speakers generated using our methods, and carry out speech recognition in order to evaluate the effectiveness of the proposed methods. Finally, in section V, we summarize the paper.

II. Manipulating Vocal Tract Length

Vocal Tract Length Normalization is generally used to remove differences in speakers' vocal tract lengths. We can achieve this by varying a certain parameter, α, called the warping coefficient. This parameter is usually estimated using a maximum-likelihood (ML) criterion [6]. By changing this parameter, we can scale the spectrum along the frequency axis f. VTLN is mathematically described as follows.

We first transform a time-dimensional waveform into a spectral form by means of a short-time Fourier transform (STFT), and then filterbank analysis is applied. During filterbank analysis, the vocal tract length operation is applied concurrently. We can express filterbank analysis as

$$O_\alpha(n) = \sum_{f=l_n}^{h_n} T_n(f)X(f), \tag{1}$$

where $O_\alpha(n)$ is a value of n_th bin when the α is applied, l_n and h_n are lowest and highest ends of n_th triangle window, $T_n(f)$ is a n_th triangle window, and $X(f)$ is a spectrum.

Frequency warping is actually applied to the central frequencies of the filterbank, which can move depending on the value of α [7]. Therefore, according to Equation (1), each of the function $T_n(f)$ and $X(f)$ is to be converted into $T_n(\hat{f})$ and $X(\hat{f})$ where $\hat{f}$ corresponds to the warped value from frequency f. Fig. 1 shows an illustration of the changing central frequencies of the filterbank. This shows how a certain value of frequency f_c is converted to another value $\hat{f}_c$.

This spectral warping is conducted through a frequency warping function. Several varieties of the function have been proposed (linear [8], piece-wise linear [9], bilinear [10] and multiple-parameter all-pass transforms [11][12]). Of these, we apply a piece-wise linear function.

Fig. 2 shows the functions related to the original frequency and the warped one. This function is proportional to the inverse of α between 0 and a certain value of the frequency, and thus we obtain warped frequency $\hat{f} = f \cdot \alpha^{-1}$. Finally, after the filterbank analysis, we extract an MFCC feature vector by applying IDCT to the logarithm of the output from the filterbank.

In our study, we use α not to normalize the feature vector, but to vary it. Instead of using an optimal α with the ML criterion, we apply an arbitrary α to operate VTL and generate pseudo feature vectors.

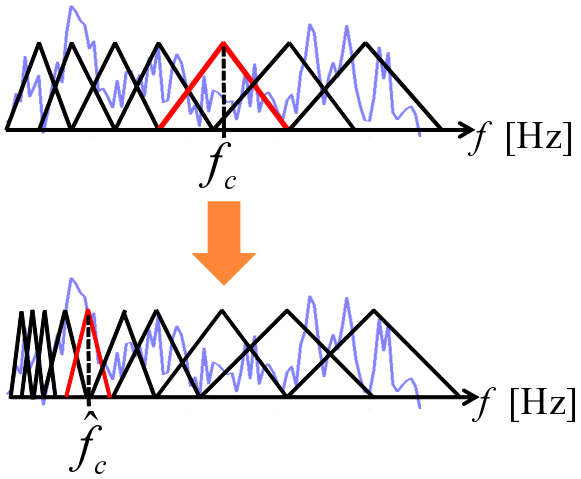

Fig. 1. Illustration of actual method used for changing central frequencies of the filterbank

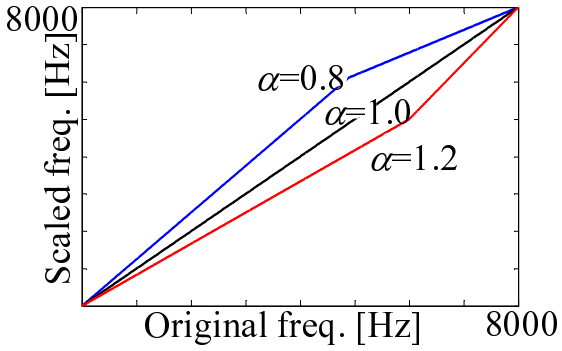

Fig. 2. Warp function used in the filterbank analysis

III. Converting Speaking Rate

The difference in speakers' speaking rates is one of the prosodic characteristics of language. Altering the speaking rate is done by scaling the waveform in the time dimension. In this study, we use the PICOLA (Pointer Interval Controlled OverLap and Add) algorithm to convert the speaking rate. Using this method, we focus on the fact that a speech wave is composed of short and cyclic waves, so we insert such waves into a speech wave or delete them from it. This method makes it possible to compress or expand a wave while keeping the pitch of voice the same, so that we can maintain the quality of voice.

In Fig. 3, we show a brief illustration of the algorithm, which shows how to compress the speech wave. From the analyzing pointer, we apply a triangular window toward the length of a pitch wave, and another window is also applied to the succeeding pitch. These two waves are defined as the A wave and the B wave. We sum A and B, to make a new wave, C. After that, we replace the wave A and B with the

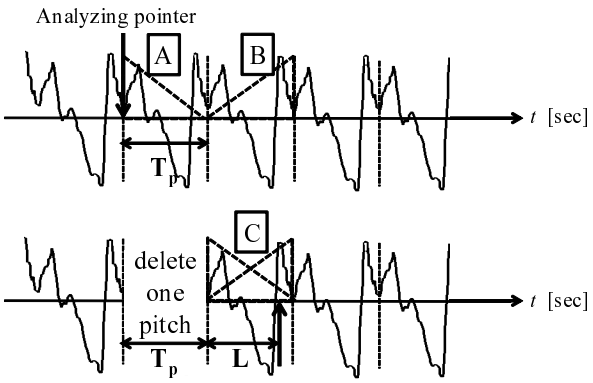

Fig. 3. Brief illustration of speech wave compression

TABLE I
NUMBER OF REAL SPEAKERS AND CORRESPONDING UTTERANCES

#Speaker	#Utterence
20	3,088
60	9,231
120	18,447
260	40,100

wave C. The reason why triangular windows are used here is to maintain continuity at the joint point of wave C. Finally, we shift the analyzing pointer to the right for distance L and then perform the same operations. We define r as the compression rate of the speech wave in the form of the following equation:

$$r = \frac{L}{T_p + L} \tag{2}$$

IV. EXPERIMENTS

A. Acoustic model conditions

Based on the two methods explained in the previous two sections, we trained acoustic models using the characteristics of pseudo-speakers' derived from a small number of real speakers. As the original training data, we used the Japanese Newspaper Article Sentences (JNAS) read speech corpus [13]. We first trained a model using 260 real speakers, which we called the large speakers' model. We also trained three other models using 20, 60 and 120 speakers, who were selected from 260 speakers. First, we first arranged 260 speakers randomly, and then we selected speakers starting from the top to the 20th, 60th, and 120th speakers, respectively. We called each of these models small speakers' models. Then, we generated pseudo-speakers from 20, 60 and 120 speakers, by changing the parameters of the vocal tract or the speaking rate. After that, we trained three models using both existing speakers and pseudo-speakers. We called each of them generated speakers' models. The numbers of existing speakers and corresponding utterances are presented in Table I.

TABLE II
TOTAL NUMBER OF SPEAKERS AND UTTERANCES USED IN 1ST TO 3RD EXPERIMENTS

#Speaker and #Utterance	Real	1st (VTL operation)	2nd (Speaking rate operation)	3rd (Combined operation)
#speaker	20	340	100	500
#utterance	3,088	52,496	15,440	77,200
#speaker	60	300	300	1,500
#utterance	9,231	46,155	46,155	230,775
#speaker	120	600	600	3,000
#utterance	18,447	92,235	92,235	461,175

B. Experimental method

We conducted four experiments in total. First, we generated pseudo-speakers based on the VTL operation. In order to evaluate the effect of the VTL operation-based method, we compared the generated speakers' model with the small speakers' models for each number of original speakers. We also compared each generated speakers' model with the large speakers' model. In order to generate pseudo-speakers' features, we applied warping coefficients from 0.8 to 1.2 to each existing speaker; 17 values (0.8, 0.825, ..., 1.2) to 20 speakers, and 5 values (0.8, 0.9, ..., 1.2) to 60 and 120 speakers.

In the second experiment, we generated pseudo-speakers based on speaking rate conversion. The value of the speaking rate is ranged from 0.75 to 1.25. The total number of given values was 5 (0.75, 0.875, ..., 1.25) for all existing speakers.

Thirdly, we combined these two methods when generating pseudo-speakers. Through this experiment, we tried to confirm these methods had any complementary effects. The number of speakers we used in each experiment is shown in Table II. As can be seen from this table, the number of speakers we used is a multiple of the number of actually existing speakers.

So far, we had not considered the effect of the total number of training speakers in all the experiments. And so, with regard to further experiments based on VTL operation, we considered the effectiveness of increasing the total number of training speakers. We varied the number of generated speakers and made another three kinds of the speakers' sets from each of the existing speakers. We increased the number of speakers as a multiple of each existing speaker (for example, from 20 speakers, we generated 140 and 340 speakers, both of which are a multiple of 20). The detailed number is shown in Table III, where "Total" speakers includes the originally existing speakers and the generated pseudo speakers. Through this experiment, we could confirm if the number of training speakers has any effect on the recognition rate.

Other experimental conditions are summarized in Table IV.

To test recognition, we used 46 speakers as test data (23 males and 23 females selected from JNAS), and none of their data were included in the training data.

C. Evaluation results

Fig. 4 shows word recognition accuracy for the first and second experiment. The horizontal line at 92.0% stands for the word accuracy of the large speakers' model. As shown, all

TABLE III
TOTAL NUMBER OF SPEAKERS AND UTTERANCES USED IN THE 4TH EXPERIMENT

#Speaker and #Utterance	Using only real speakers	Using both real and pseudo speakers	
		Trial1	Trial2
#Speaker	20	140	340
#Utterance	3,088	21,616	52,496
#Speaker	60	300	420
#Utterance	9,231	46,155	64,617
#Speaker	120	600	840
#Utterance	18,447	92,235	129,129

TABLE IV
EXPERIMENTAL CONDITIONS

Training DB	JNAS
Feature vectors	12 MFCC + 12 ΔMFCC +ΔPower
decoder	Julius-4.1.5 [14]
Acoustic model structure	Speaker-independent triphoneHMM 3000states, 16mixtures per state
Language model	3-gram model (paper articles to the amount of 75months)
Number of words in dictionary	21,322
Evaluation data	IPA-98-Testset (Subset of JNAS, 23 males and 23 females)
Number of evaluation speech	200 (100 by males and 100 by females)

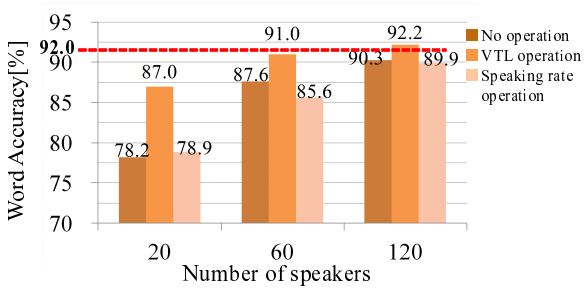

Fig. 4. Word accuracy in the first and second experiments (VTL operation and speaking rate operation)

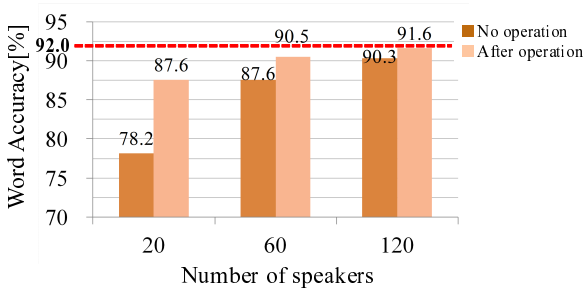

Fig. 5. Word accuracy in the third experiment (Combining VTL with speaking rate operation)

of generated speakers' models based on VTL operation show better performance than the small speakers' models. The word accuracy of the generated speakers' model using 120 speakers is a bit higher than that of large speakers' model. The result shows that VTL operation-based method can express a lot of speech variations and thus contribute to improving recognition accuracy.

In contrast, as shown in the second experiment, the effect of converting the speaking rate is very small. We assume that such poor result are caused by the method used to convert the speaking rate. In the PICOLA algorithm, as we saw in section III, pitch waves were maintained. Only the speaking rate itself actually changed. When the speaking rate changes, the durations of the utterances also changes, and thus the transition probabilities in HMMs are mainly affected. The PICOLA algorithm is able to maintain speech quality well, and spectral information was only slightly changed. Thus, the feature distribution parameters are not greatly varied. So it is not effective to express spectral variations, and therefore the experiment showed poor results.

The result of the third experiment is shown in Fig. 5. As well as in the previous experiments, the horizontal line stands for the word accuracy of large speakers' model. As shown in this graph, we could not obtain complementary effects through the combination of two of proposed methods. We can see better performance than with the method of converting speaking rate alone, however, these results show slightly worse performance than using the VTL operation-based method. As in the second experiment, the PICOLA algorithm is not effective in this experiment.

Finally, we show the results of the fourth experiment in Fig. 6. In contrast to the previous experiments, the horizontal axis shows the total number of used speakers. From this figure, we can see that word accuracy differs only a little as a result of changing the total number of speakers. However, we can also assume that all the models have their own upper limits on the total number of speakers. These results show that the number of speakers used for training acoustics models has an effect on word accuracy, and that there is a optimal number of real speakers needed for VTL operation to be most effective.

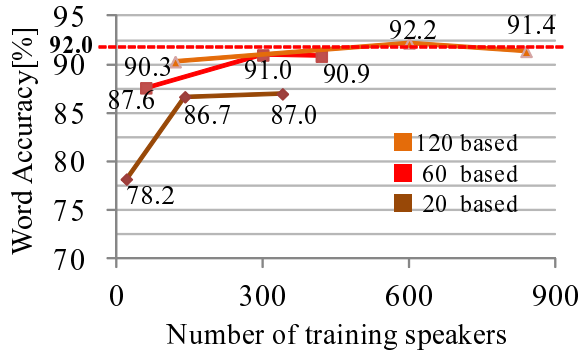

Fig. 6. Word accuracy in the fourth experiment (VTL operation changing the number of the speakers)

V. Conclusions

In this paper, we proposed generative training methods for acoustic models by explicitly manipulating speaker characteristics and speaking styles. We generated pseudo-speakers by altering speakers' vocal tract length and their speaking rate. We found that by training acoustic models not only with real speakers, but also using pseudo-speakers created using the VTL operation as training data, the models were more robust to unknown speakers. On the other hand, speaking rate conversion was not effective in generating realistic pseudo-speakers. Additionally, combining the VTL operation with the speaking rate conversion was not effective either, and we could not obtain any complementary effects. However, it was implied that the number of speaker samples has an effect on raising word accuracy.

We imagine that creating pseudo-speakers by converting the speaking rate of actual speakers may be more effective for recognition of spontaneous speech than read speech, since this method may be useful for expressing prosodic characteristics.

In the future, we will consider other ways to represent speaker variation, when training acoustic models in order to achieve robust recognition of spontaneous speech, such as that found in the Corpus of Spontaneous Japanese (CSJ) [15]. We will also explore using training data from training data recorded in another environment, like JNAS, for this purpose.

References

[1] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," in Proc. IEEE Trans. Speech Audio Processing, vol. 2, pp. 291–298, 1994.

[2] M.J.F. Gales, "Maximum likelihood linear transformations for HMM based speech recognition," in Computer Speech and Language, vol. 12, no. 2, pp. 75–98, 1998.

[3] V.V. Digalakis, D. Rtischev, and L. G. Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures," IEEE Trans. Speech and Audio Processing, vol. 3, No. 5, pp. 357–366, 1995

[4] A. Itoh, S. Hara, N. Kitaoka and K. Takeda, "Acoustic Model Training Using Pseudo-Speaker Features Generated by MLLR Transformations for Robust Speaker-Independent Speech Recognition," in Proc. IEICE Trans. Inf. Syst., (to appear), 2012.

[5] S. Jan, C. Petr, and Z. Jindrich, "MLLR transforms based speaker recognition in broadcast streams," in Cross-Modal Analysis of Speech, Gestures, Gaze and Facial Expressions, pp. 423–431, 2009.

[6] J. McDonough, T. Shaaf and A. Waibel, "Speaker adaptation with all-pass transforms," in Proc. Speech Commnication, vol. 42, pp. 75–91, 2004.

[7] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," in Proc. ICASSP, pp.353–356, 1996.

[8] S. Wegmann, D. McAllaster, J. Orloff and B. Peskin, "Speaker normalization on conversational telephone speech," in Proc. ICASSP, vol. I, pp. 339–341, 1996.

[9] L. Lee and R. Rose, "A frequency warping approach to speaker normalization," in Proc. IEEE Trans. Speech Audio Processing, vol. 6(1), pp. 49–60, 1998.

[10] J. McDonough, F. Metze, H. Soltau, and A. Waibel, "Speaker compensation with sine-log all-pass transforms," in Proc. ICASSP, pp. 369–372, 2001.

[11] M. Pitz and H. Ney, "Vocal Tract Normalization as Linear Transformation of MFCC," in Proc. IEEE Trans. Speech Audio Processing, pp. 930–944, 2005.

[12] S. Umesh, A. Zolnay and H. Ney, "VTLN using analytically determined linear transformation on conventional MFCC" in Proc. IEEE Trans. Speech Audio Processing, pp. 1573–1584, 2012.

[13] K. Itou, M. Yamamoto, K. Takeda, T. Takezawa, T. Matsuoka, T. Kobayashi, K. Shikano, S. Itahashi, "JNAS:Japanese speech corpus for large vocabulary continuous speech recognition research," in The Jour nal of the Acoustical Society of Japan (E), Vol. 20, No.3, pp.199–206, 1999.

[14] T. Kawahara and A. Lee, "Open-source speech recognition software Julius," in JSAI, vol. 20, no. 1, pp. 41–49, 2005.

[15] S. Furui, K. Maekawa, and H. Isahara, "A Japanese national project on spontaneous speech corpus and processing technology," in Proc. ASR2000, pp. 244–248, 2000.