ISCA Archive
http://www.isca-speech.org/archive

International Symposium on Chinese Spoken Language Processing (ISCSLP 2002)
Taipei, Taiwan
August 23-24, 2002

ENHANCING THE STABILITY OF SPEAKER VERIFICATION WITH COMPRESSED TEMPLATES

WEN Xue

Department of Electronic Engineering,
Tsinghua University, Beijing
wenx@public2.east.net.cn

LIU Runsheng

Department of Electronic Engineering,
Tsinghua University, Beijing
lrs-dee@mail.tsinghua.edu.cn

ABSTRACT

Time-domain template compression is an effective means to reduce storage and computation complexity of speaker verification systems based on template matching. Yet this compression may cause severe performance deterioration. In this paper we propose a frame-level verification method to cut down this deterioration. A frame discrimination procedure is then introduced to further improve the verification performance. These methods add only a little to the computation and storage load, yet effectively enhance the verification stability against template compression. With these improvements, we have cut down the deterioration by more than 2/3, and have gained a verification EER of 2.35% with templates compressed at an 8:1 rate.

Keywords: speaker verification, dynamic time warping

1. INTRODUCTION

Template matching methods are used for text-dependent speaker verification tasks on low-cost speech processing devices for their simplicity and relatively good performance with limited training data[1,2]. Usually a sequence of vectors generated from training speech is stored as a template during training session. In the verification phase, a feature vector sequence generated from test speech is matched to the template to produce a verification score, usually a distance measure[3]. Even with this simplicity, there may still be some crucial issue on the limits of system storage and computation capacity, especially for low-end systems. Time-domain template compression, for example, by segmental k-means method, may be helpful to reduce storage and computation requirements, with a side effect of performance deterioration[4]. Here we propose an improved framework on the scoring of conventional DTW method, which significantly cuts down the level of performance deterioration caused by template compression.

2. THE METHOD

Conventionally an overall distance between the template and test vector sequence is computed by averaging frame-level distances. Suppose the template **X** contains a sequence of T' vectors, and the test vector sequence **Y** contains T vectors, $\mathbf{X}=\{\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots\cdots, \boldsymbol{X}_{T'}\}$, $Y=\{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \cdots\cdots, \boldsymbol{Y}_T\}$. Then the overall distance between **X** and **Y** is computed as

$$D(\mathbf{X},\mathbf{Y}) = \frac{1}{T}\sum_{t=1}^{T} d(t) = \frac{1}{T}\sum_{t=1}^{T} d(\boldsymbol{X}_{Q(t)}, \boldsymbol{Y}_t) \qquad (1)$$

where $Q(t)$ is an monotonic alignment function selected through a dynamic time warping (DTW) procedure to minimize $D(\mathbf{X},\mathbf{Y})$, with $Q(1)=1$, $Q(T)=T'$, and $d(\boldsymbol{X},\boldsymbol{Y})$ is a measure of distance between two frames (vectors). Verification decision, either "accept" or "reject", is made by comparing $D(\mathbf{X},\mathbf{Y})$ to an a priori threshold.

In our new framework a verification decision is made for each frame of test data. Namely, after the DTW alignment, we compare each frame-level distance $d(t)$ (where t=1, 2, ……, T) to some a priori threshold *Th*, and tag each frame of test data with a verification result such as "accept" or "reject". After all the frames of test speech have been tagged, a final decision is drawn

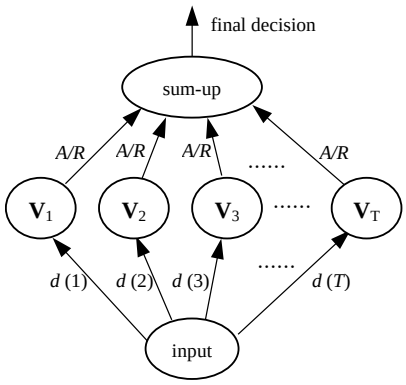

Figure 1
Improved speaker verification framework

on all these frame-level results.

Figure 1 shows this framework. The information contained in each single frame is limited, so each frame-level verification gives only a rough result, which is relatively less sensitive to subtle changes in template vectors. We can thus expect the final decision drawn on these results be more stable than that of the conventional framework.

2.1 Frame-level Verification

For frame-level verifications, a universal a priori threshold is used for all frames, regardless of the speaker or the text of speech. The selection of the threshold is based on statistics. On a separate set of speech data, we perform the conventional DTW alignments in great amounts, covering as many speakers and text contents as available. Each alignment is done on two utterances of the same text, either from one same speaker (client test) or from two speakers (imposter test). Then we calculate the overall distribution of frame-level distances we have gained as a by-product of these alignment procedures. One distribution, $P_C(d)$, is calculated for client tests and another, $P_I(d)$, for imposter tests. The results are shown in Figure 2.

A likelihood ratio $LR(d)$ is calculated as follows:

$$LR(d) = \frac{P_C(d)}{P_I(d)} \tag{2}$$

It is observed that within the main interval of d's distribution $LR(d)$ is monotonically decreasing. The point $\hat{d}$ where $LR(d)$ =1 is used as the threshold for frame-level verification. If a frame of test speech matches the template frame it is aligned to with a distance d that is smaller than $\hat{d}$ (so that $LR(d)$ >1), it is tagged with "client"; if not, it is tagged with "imposter".

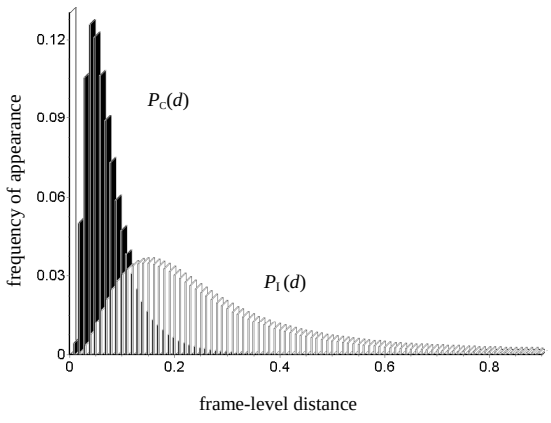

Figure 2

Distributions of frame-level distances

2.2 Frame Discrimination

As has been mentioned, frame-level verifications give only rough results, on which the final decision will be drawn. The reliability of these results is questionable. In general, different frames of speech may have different capacities in distinguishing a speaker. For each speaker there may be a specific phoneme set that is especially effective in distinguishing him/her from others. As to frame-level verification, it is reasonable that a more discriminative frame of speech will give a more reliable result. The verification results made on the least discriminative frames are error-prone and may lead to final verification failure. A procedure to de-emphasize their effects on the final decision will be necessary to improve the overall verification performance.

The first step here is to detect the least discriminative frames. For a frame from test data with a distance d to its matching template frame, by assuming an equal prior distribution of client and imposter tests (i.e. both assumed to be 1/2), we can estimate their posterior distribution as follows:

$$P(\mathrm{I} \mid d) = \frac{1}{1 + LR(d)} \tag{3}$$

$$P(\mathrm{C} \mid d) = \frac{LR(d)}{1 + LR(d)} \tag{4}$$

Note that (3) gives the false acceptance rate when a decision of acceptance is made, and (4) gives the false rejection rate when a decision of rejection is made. It is easy to verify that the threshold $\hat{d}$ we have selected assures the frame-level verification error rate $E_F(d)$ to be the smaller of the two

$$E_F(d) = \min\left\{\frac{1}{1 + LR(d)}, \frac{LR(d)}{1 + LR(d)}\right\} \tag{5}$$

$E_F(d)$ reaches its maximum 0.5 when $LR(d)$=1.

We use $E_F(d)$ as a measure of each frame's reliability. A frame with an $E_F(d)$ greater than some threshold C ($0<C<1/2$) is thought not discriminative enough to give a reliable result of frame-level verification for final decision, and that result will be de-emphasized in the final verification phase. From $E_F(d)>C$ ($0<C<1/2$) we have

$$\frac{C}{1-C} < LR(d) < \frac{1-C}{C} \tag{6}$$

Thus, instead of calculating $E_F(d)$ for each frame, we select a pair of thresholds $\hat{d}_1$ and $\hat{d}_2$, $\hat{d}_1 < \hat{d} < \hat{d}_2$, and

$$LR(\hat{d}_1) = \frac{1-C}{C}, \quad LR(\hat{d}_2) = \frac{C}{1-C} \tag{7}$$

A frame with a $d(t)$ within the interval $(\hat{d}_1, \hat{d}_2)$ is thought error-prone.

2.3 Final Decision

The last step is easy. Each frame tagged "client" in frame-level verification is assigned a score –1 and each tagged "imposter"

assigned +1, except that all the "unreliable" frames are assigned 0. These new scores are averaged into an overall score, which is compared to a threshold Th to give out a final decision.

3. RESULTS

Our client speaker set contains 13 speakers, all male, aged between 15 and 25. Totally 280 utterances are taken from each client, 10 for each word in a 28-word set common to all speakers. The 28 words are composed of three groups: 8 short words each containing 4 or 5 Chinese characters, 10 short strings each containing 4 Chinese digits, and 10 long strings each containing 8 Chinese digits. All these characters and digits are monosyllables. The imposter set contains 40 speakers, all male, aged between 15 and 25. 140 utterances are taken from each imposter, 5 for each word in the word set described above.

All utterances are sampled at 16kHz. Each frame contains 256 samples, with 128 of them overlapping with its preceding frame. For feature vectors, 14 linear predictive cepstral coefficients (LPCC14) are extracted from each frame of speech.

All the following tests use template pairs, i.e. two client utterances of the same word are used for training a client password, each into a template separately. For tests on conventional method, the distance between test speech and a template pair is calculated as the smaller of the distances between test speech and each of the templates. For tests on our proposed method, the template with a small distance to test speech is used for calculating $d(t)$, t=1, 2, ……, T.

For each template pair, 8 client utterances of the password not involved in training the pair are used for client test, and 20 imposter utterances of the password are used for imposter tests. We always run tests with correct passwords.

Equal error rates (EER) are used as the measure of performance

3.1 Test without Template Compression

This test compares the basic performances of the conventional DTW method and our method. The test is run separately on each of the three groups of words and on the whole word set. Note that each group has its own threshold at EER and the overall EER is calculated by running a test with its own specific threshold, not by averaging the three group EER's.

We use C=1/6 in (7) for the selection of $\hat{d}_1$ and $\hat{d}_2$, which gives a comparatively good performance with and without template compression.

Table 1 Result without Template Compression

Group	EER (%) (Standard DTW)	EER (%) (Our Method)
Short Words (SW)	2.28	2.11
Long Strings (LS)	1.34	1.27
Short Strings (SS)	2.19	2.08
Overall	1.93	1.81

Table 1 shows the result. It is clear that the performance of our method is well comparable to that of the standard DTW method.

3.2 Test with Template Compression

Time-domain template compression is done by segmenting the template vector sequence and then substituting each segment of the sequence with one vector, which is the average of the vectors that segment contains. In our tests a compression rate of 8 is used. The length of each segment is limited by putting a threshold on the variation coefficient of its frames, which helps to preserve more information after compression.

Table 2 shows the result. C=1/6 is used in (7) for the selection of $\hat{d}_1$ and $\hat{d}_2$.

From table 2 we see that the verification system using traditional DTW method suffers seriously from template compression, with an increase in overall EER by 102.1%. Our new method performs much more robustly with this compression, with an increase in overall EER by 29.8%.

Table 2 Result with Template Compression

Group	Standard DTW		Our method	
	EER (%)	Increase (%)	EER (%)	Increase (%)
SW	4.61	102.2	2.79	32.2
LS	2.47	84.3	1.61	26.8
SS	4.68	113.7	2.66	27.9
Overall	3.90	102.1	2.35	29.8

3.3 Supplemental Tests

These tests are designed to give an insight into the new method. Two improvements, namely frame-level verification and frame discrimination, are made on conventional DTW method to bring out the new method. Here we test the effects of these two improvements separately to see what each has done to enhance the system.

Test on frame-level verification is done simply by running the test on our new method while setting C=1/2 in (7). To test the effect of frame discrimination alone, we calculate verification scores with (8), which removes all frames that are detected "unreliable" from (1):

$$\hat{D}(\mathbf{X},\mathbf{Y}) = \sum_{\substack{t=1 \\ d(t)\notin(\hat{d}_1,\hat{d}_2)}}^{T} d(t) \Bigg/ \sum_{\substack{t=1 \\ d(t)\notin(\hat{d}_1,\hat{d}_2)}}^{T} 1 \qquad (8)$$

Table 3 and table 4 show the results. It's clear that neither frame-level verification nor frame discrimination by itself improves the performance when compression is not used, but frame discrimination does help to reduce the EER of a system with frame-level verification by 10%. When template compression is used, frame-level verification on its own has made great

improvement on performance. The effect of frame discrimination is not as significant, but again, this procedure improves a system with frame-level verification significantly, this time by 16%.

Table 3 Result with Frame-level Verification only

Group	EER (%) without Compression	EER (%) with Compression	Increase (%)
SW	2.29	3.13	36.7
LS	1.37	2.02	47.4
SS	2.28	3.18	39.5
Overall	2.00	2.73	36.5

Table 4 Result with Frame Discrimination only

Group	EER (%) without Compression	EER (%) with Compression	Increase (%)
SW	2.24	4.33	93.3
LS	1.30	2.39	83.8
SS	2.34	4.24	81.2
Overall	2.02	3.58	77.2

Figure 3 compares the four systems we have tested. These are conventional DTW system (DTW), DTW system with frame discrimination (DTW-FV), DTW system with frame-level verification (DTW-FD), and DTW system with both the two (DTW-FV/D), which is our proposed new system.

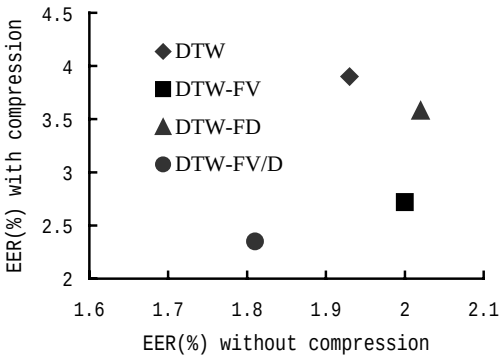

Figure 3

Overall Performance Comparison

4. CONCLUSION

Template compression is an effective means to reduce storage and computation requirements of template matching procedures, which may be a crucial issue in on-chip systems. Yet the use of such compression has been limited because of performance considerations. For text-dependent speaker verification tasks, the EER is doubled with an 8 to 1 compression. Yet this deterioration not only comes from the information loss caused by compression, but also comes as a result of the internal instability of conventional DTW scoring method. With some adjustment on the traditional DTW framework, i.e. a frame-level verification scheme, we have made this verification system much more robust with compressed templates. Also we use a frame discrimination procedure to effectively enhance the performance of this improved system. These procedures add little to storage and computation complexity of the whole system, and give a possible solution to the performance problem with compressed models.

5. ACKNOWLEDGEMENTS

This project is supported by National Foundation on Natural Science (No.69975007) and National 863 High Technology Projects (No.863-306ZD13-04-6).

6. REFERENCES

[1] Atal B S. *Automatic Recognition of Speakers from Their Voices*. Proceedings of IEEE, 1976, vol. 64, no.4: 460-475.

[2] Yu K, Mason J, Oglesby J. *Speaker Recognition Using Hidden Markov Models, Dynamic Time Warping and Vector Quantisation*. IEE Proc. Vis. Image Signal Process. , October 1995, vol. 142, no.5: 313-318.

[3] Rabiner L, Juang B H. *Fundamentals of Speech Recognition*. Prentice Hall, 1993, Chapter 4 & 5.

[4] Shi Y, Liu J, Liu R. Single-Chip Speech Recognition System Based on 8051 Microcontroller Core. IEEE Tran. on Consumer Electronics, 2001, vol.47 no.1: 149-154.