

# Text Independent Speaker Recognition Using Speaker Dependent Word Spotting

Hagai Aronowitz<sup>1</sup>, David Burshtein<sup>2</sup> and Amihod Amir<sup>1,3</sup>

<sup>1</sup>Department of Computer Science, Bar-Ilan University, Israel

<sup>2</sup>School of Electrical Engineering, Tel-Aviv University, Israel

<sup>3</sup>College of Computing, Georgia Tech, USA

## Abstract

This paper is motivated by the fact that text dependent speaker recognition is inherently more accurate than text independent speaker recognition. In this work we assign models to frequent words spoken by a speaker and spot them in a test call. In this way, text-dependent speaker recognition technology can be used for text independent tasks. The approach we take is to use DTW (Dynamic Time Warp) word spotting to find words in the test that resemble words in the train set. Results on the SPIDRE corpus show that using a combined DTW spotter based system and a GMM system improves performance significantly. For very low false acceptance rate (0.1%) misdetection was reduced from 32.2% to 23.3% (28% reduction). For low false acceptance rate (1%) misdetection was reduced from 28.9% to 21.1% (27% reduction).

## 1. Introduction

Speaker recognition aims to recognize the identity of a speaker given an audio input with knowledge of the lexical content (text-dependent) or without that knowledge (text-independent). A common text-dependent recognition task is vocal password access control in which each user has a private predefined vocal password. Text-independent recognition tasks such as call routing usually do not make any assumptions on the lexical content of the call. It is well known that given the same experimental conditions (amount of data for training/testing, noise conditions, etc.) text-dependent speaker recognition is more accurate than text-independent recognition. This fact can be better realized by noting the fact that people can sometimes recognize a speaker very rapidly by only hearing his "hello" or some other familiar word.

The motivation for using text-dependent technology for the text-independent task was demonstrated in [1] where a large vocabulary continuous speech recognizer (LVCSR) was used to convert the text-independent task to a text-dependent task. However, the use of a LVCSR has some drawbacks. Training a LVCSR requires many resources (a large transcribed corpus, LVCSR software) which are often not available. Furthermore, running a LVCSR is computationally complex. Finally, the portability and robustness of LVCSR systems are currently limited.

In this work we exploit the fact that the distribution of words in a spoken call is such that a reasonable small set of words covers a considerable percent of the spoken words, i.e., there are some very frequent words. This fact leads to an efficient strategy: instead of recognizing tens of thousands of words, just focus on frequent words (few hundreds) and ignore

the rest of the words. The text-dependent algorithm is therefore applied only to frequent words, and the rest of the audio is modeled only by text-independent methods. Another fact exploited is that speaker dependent word spotting is usually more accurate than speaker independent word spotting, hence speaker dependent word spotting is used to recognize the frequent words, potentially more accurate than LVCSR. In this paper we chose to use dynamic time warp (DTW) [2] for speaker dependent word spotting and the same DTW score is used as a text-dependent speaker recognition score. The outcome of the DTW spotter is a set of putative frequent words appearances and their corresponding scores. These text-dependent speaker resemblance scores need to be fused and combined with a text-independent score of the rest of the call (where no frequent words are spotted).

The organization of this paper is as follows: the DTW based speaker recognition system is presented in Section 2. Section 3 describes the experimental corpus, the experiments and the results. Finally, section 4 presents conclusions and possibilities for future work.

## 2. DTW based speaker recognition

In this section we describe the proposed DTW based speaker recognition system. After presenting the DTW based system we describe a method to combine the DTW based system and the text-independent GMM system.

### 2.1. Outline of the proposed DTW based system

#### Training phase:

1. Compose a set of target words.
2. For each target speaker training call, mark all occurrences of words from the set of target words. Each occurrence is considered as a different template.
3. For every template, spot all reasonable matches in development set.
4. Tune template-dependent thresholds according to state 3.
5. For every development-set call, calculate a fused DTW score.
6. Tune a speaker-dependent normalization function for fused DTW scores according to stage 5.

#### Testing phase:

For every call and for every target speaker:

1. For every template of the target speaker, spot all matches in development set.
2. Filter all spotted matches according to template-dependent thresholds.
3. Calculate a fused DTW score.

4. Normalize fused DTW score according to speaker dependent normalization function.

## 2.2. DTW front-end

The front-end of the recognizer consists of calculation of Mel-frequency cepstrum coefficients (MFCC) according to the ETSI standard [3]. An energy based voice activity detector is used to locate speech segments and the cepstral mean of the speech segments is calculated and subtracted. The final feature set is 13 cepstral coefficients + 13 delta cepstral coefficients extracted every 10ms using a 20ms window.

## 2.3. Composing the set of target words

The set of target words is language specific and can be domain specific and even speaker specific. The set of target words can be selected automatically from acoustic data only, from acoustic data transcribed manually or by an ASR system, or from transcriptions only. In this paper the set of target words is composed of all the words that are more frequent in the transcript of the corpus than a predefined threshold (288 words were chosen). All occurrences of words from the target set in the training set are used for DTW spotting.

## 2.4. DTW spotting

After finding all the templates in the training set (either automatically or by exploiting available transcript) each template is searched in a given test call for possible matches. A match is defined by a low score of a DTW matcher. A detailed description of the DTW algorithm can be found in [2]. The original DTW algorithm was modified in order to be used efficiently as a word spotter. A template of cepstral vectors  $X_1, \dots, X_m$ , can be spotted in a sequence  $Y_1, \dots, Y_n$  by filling a table  $T$  where  $T_{ij}$  represents the accumulated distance between the prefix of the template  $X_1, \dots, X_i$  and the best suffix of  $Y_1, \dots, Y_j$ .  $T$  can be computed by using the recurrence equation:

$$T_{i,j} = \min \left\{ \begin{array}{l} T_{i-1,j-1} + d_{i,j} \\ T_{i-1,j-2} + \frac{d_{i,j} + d_{i,j-1}}{2} \\ T_{i-2,j-1} + d_{i,j} + d_{i-1,j} \end{array} \right\} \quad (1)$$

In equation (1)  $d_{ij}$  is the local normalized distance between the cepstral vector  $X_i$  and the cepstral vector  $Y_j$ . For simplicity,  $d_{ij}$  was chosen in this paper as the Euclidean distance and no normalization was used. The final matching score is normalized by the length of the template ( $m$ ).

The DTW distance as defined in equation (1) actually projects a segment of  $Y$  onto the template  $X$  which assures that the final length normalization does not affect the optimality of the search. The DTW spotter finds all matches of template  $X$  in  $Y$  in time  $O(nm)$ .

## 2.5. DTW scores normalization

Given a test call with a set of DTW scores corresponding to templates of a target speaker, all scores should be normalized and fused to a single score. Ideally each DTW score should be transformed to a log-likelihood ratio score. Unfortunately, the sparseness of positive development data (target speaker calls)

makes it hard to estimate the distribution of the scores of a given template on target data; therefore each DTW score is converted to a true/false decision where true means that the putative hit is considered a correctly detected word of the target speaker with relatively high confidence. More specifically, for a putative hit for template  $W$  and score  $S$ , the score  $S$  is compared to a template specific threshold  $\Theta_W$ , and only if the score is lower than the threshold it is assigned the value 'true'.  $\Theta_W$  is computed on a development set (with only non-target conversations) and is tuned in such a way that the number of putative hits for template  $W$  on the development set is very small and identical for all the templates.

## 2.6. Fusing normalized DTW scores

Given a test call with a set of normalized DTW scores corresponding to a target speaker, all scores should be fused to a single score. Assuming all putative hits surviving threshold cutoff are distributed independently, the probability of observing  $n_i$  putative hits for template  $i$ , for  $i=1, \dots, N_T$ , where  $N_T$  is the number of templates is:

$$\Pr(n_1, \dots, n_{N_T} | True) = \prod_{i=1}^{N_T} \Pr(n_i | True) \quad (2)$$

and

$$\Pr(n_i | True) = \binom{len}{n_i} \left( \frac{\alpha_i}{1 - \alpha_i} \right)^{n_i} (1 - \alpha_i)^{len} \quad (3)$$

where  $len$  is the length of the test call, and  $\alpha_i$  is the probability of template  $i$  to match at an arbitrary frame of a target (true) call. Writing the corresponding equations for non-target calls and combining with equations (2,3) leads to equation (4):

$$\log \left( \frac{\Pr(n_1, \dots, n_{N_T} | True)}{\Pr(n_1, \dots, n_{N_T} | False)} \right) = \sum_{i=1}^{N_T} \left( n_i \log \frac{\alpha_i (1 - \beta_i)}{\beta_i (1 - \alpha_i)} + len \log \frac{1 - \alpha_i}{1 - \beta_i} \right) + \sum_{i=1}^{N_T} n_i \log \frac{\alpha_i (1 - \beta_i)}{\beta_i (1 - \alpha_i)} + len \delta_{speaker} \quad (4)$$

In (4)  $\beta_i$  is the probability of template  $i$  to match at an arbitrary frame of a non-target (false) call, and  $\delta_{speaker}$  is a speaker dependent constant. The procedure of tuning the thresholds for the templates implies constant  $\beta_i$ :  $\beta_i = \beta$ .  $\alpha_i$  is definitely different between templates (and reflects the quality of a template to discriminate between the target speaker and non-target speakers). Again, it is hard to estimate  $\alpha_i$  because of the sparseness of positive development data and in this paper  $\alpha_i$  is assumed to be constant:  $\alpha_i = \alpha$  which leads to the simplified equation (5):

$$\log \left( \frac{\Pr(n_1, \dots, n_{N_T} | T)}{\Pr(n_1, \dots, n_{N_T} | F)} \right) = w \sum_{i=1}^{N_T} n_i + len \delta_{speaker} \quad (5)$$

The consequence of equation (5) is that for each test call the log likelihood ratio for a target speaker is linearly related to the number of putative hits scoring better than their corresponding thresholds and should be compensated by the length of the call. The inaccuracy of the independence assumption and the fact the true values of  $\{\alpha_i\}$  are not really constant implies that the resulting scores should be further normalized in order to achieve good performance. The approach we have taken is to convert each score to a false acceptance rate. A score  $S$  is normalized (speaker dependently) to the score  $fa-rate$  if and only if accepting all scores better than  $S$  (for a specific speaker) results in a false acceptance rate of  $fa-rate$  for that speaker. This is done by using the development non-target data for calculating the false acceptance rate for each speaker score  $S$ .

### 2.7. Combining GMM scores and fused DTW scores

The DTW based speaker identification system focuses only on rare segments of speech where it detects a good match of a frequent word. The rest of the speech is not modeled by the DTW system. Therefore the DTW scores should be combined with the scores of a system that models the entire speech, such as a GMM based systems [4]. Theoretically the right way to combine DTW and GMM scores can be derived from the following equation:

$$\log\left(\frac{\Pr(x_1, \dots, x_n | T)}{\Pr(x_1, \dots, x_n | F)}\right) = \text{DTW-LLR} + \text{GMM-LLR} \quad (6)$$

In equation (6) DTW-LLR is the log-likelihood ratio of the DTW system and GMM-LLR is the log-likelihood ratio of the GMM system omitting the frames modeled by the DTW system (the frames covered by the matched hits) which are few compared to the rest of the frames. Therefore GMM-LLR can be approximated by the log-likelihood ratio of a regular GMM system. Unfortunately, GMM based systems use inaccurate assumptions (such as the frame independence assumption), as does the DTW system. Furthermore, equation (6) assumes independence between the DTW and the GMM scores. In this paper the DTW-system was combined with a GMM system by exploiting the fact that reasonable misdetection rate can be achieved by the DTW-system with very low false acceptance (0.1% and less). The DTW scores are normalized as described in the previous subsection to a predicted false-acceptance rate score and so are the GMM scores. The final combined score is the DTW-score if the DTW-score is better (lower) than 0.1; otherwise it is set to the GMM-score. This combination scheme is tuned to a low false-acceptance region in the DET curve. Other combination schemes can be used for tuning the combined systems to other regions in the DET curve.

## 3. Experimental results

### 3.1. The SPIDRE corpus

Experiments were conducted on the SPIDRE corpus [5] which is a subset of the Switchboard-I corpus. The SPIDRE consists of 45 target speakers, four conversations per speaker, and 100 2 sided non-target conversations. All conversations are about 5 minutes long and are all from land-line phones with mixed

handsets. The SPIDRE corpus is manually transcribed. The 100 non-target conversations were divided to the following subsets: fifty two-sided conversations were used as training and development data, and the other fifty two-sided conversations were used as test data. The four target conversations per speaker were divided randomly to two training conversations and two testing conversations, therefore some of the tests are in matched handset condition and some are in mismatched handset condition. The second side of the training target conversations was used as additional development data, and the second side of the testing target conversations was used as additional non-target testing data.

### 3.2. The baseline GMM system

The baseline GMM system in this paper was inspired by the GMM-UBM system described in [4, 6]. The front-end of the system is identical to the DTW system: ETSI based MFCC, 13 cepstral coefficients + 13 derivatives extracted every 10ms using a 20ms window, followed by energy based non-speech removal and CMS. Initially a gender independent UBM is trained using 100 non-target conversation sides (about 8 hours of speech + non-speech). Target speakers were trained using MAP adaptation. The model order was optimized resulting in an order of 2048 Gaussians. A fast scoring technique was used in which only the top 5 highest scoring Gaussians are rescored using the target models [6]. In the verification stage, the log likelihood of each conversation side given a target speaker is divided by the length of the conversation and normalized by the UBM score. The resulting score is then normalized to a false acceptance rate using the same method described in subsection 2.6 which is actually a non-parameterized generalization of z-norm [4].

The DET curve of the GMM system is presented in Figure 1. The EER of the GMM system is 8.9%.

### 3.3. DTW performance

The DET curve of the DTW system is presented in Figure 1. It can be seen that the DTW system performs quite close to the GMM system (though not as good) and in general its misdetection rate is about 7% absolutely higher than the GMM system.

The EER of the DTW system is 10.6%.

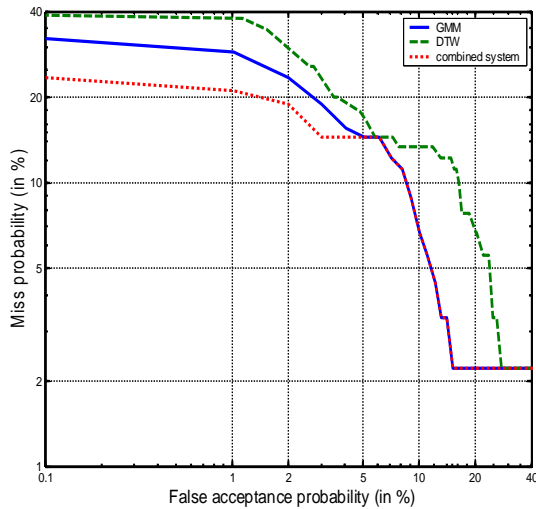
### 3.4. Combined system performance

The GMM and the DTW systems were combined as described in subsection 2.7. As mentioned previously, the combination algorithm must be targeted to a specific false-acceptance rate. Experiments have been done to check the combination algorithms when targeted to 0.1%, 1%, 5% and 10% false acceptance. The DET curve for a combination algorithm targeted to 0.1% false acceptance is presented in Figure 1. It can be seen that the combination algorithm improves the GMM performance by about 9% absolutely in the 0.1-1% false acceptance region, but has no improvement in the region 5% false acceptance and above. Other experiments show small improvement for the 5%-10% false acceptance region when using a suitable combination algorithm. Tables 1 and 2 give the DCF [7] (detection cost function) at 0.1% and 1% false acceptance and the relative improvement (reduction) of the DCF according to the baseline. The DCF definition used

in this paper is a weighted sum of both misdetection and false acceptance probabilities and is defined as:

$$DCF = P_{\text{miss}} + 9.9 \times P_{\text{fa}} \quad (7)$$

Tables 1,2 indicate a 20-27% reduction in the cost function using the combined system compared to using the GMM baseline system.



**Figure 1:** DET curve comparing the baseline GMM system to the DTW system and the combined GMM+DTW system.

System	DCF @ FA=0.1%	Improvement
Baseline (GMM)	0.332	-
DTW	0.398	(-20.2%)
Combined system	0.243	26.8%

**Table 1:** Performance of the evaluated systems at very low false acceptance (0.1%).

System	DCF @ FA=1%	Improvement
Baseline (GMM)	0.388	-
DTW	0.477	(-22.9%)
Combined system	0.310	20.1%

**Table 2:** Performance of the evaluated systems at low false acceptance (1%).

### 3.5. Complexity

The aim of this paper is a qualitative test of a new concept - that of exploiting known dependencies for speaker recognition. At this stage, the algorithm's speed was of minor concern. Indeed, running the DTW spotter is more CPU consuming than running the GMM system. The reason for this is that there are many templates that must be spotted in

parallel. This issue will be addressed in future work and has been given a reasonable solution by using the GMM system as a filter and running the DTW system only on conversations that got reasonable GMM scores. Using the GMM system as a filter did not cause any degradation in recognition accuracy.

## 4. Conclusions

In this paper we have presented a text independent speaker identification system which utilizes information in the word level by using text-dependent technology, more specifically DTW speaker-dependent word spotting. The use of this approach improved speaker identification performance on a telephone landline mixed handset corpus significantly. For very low false acceptance rate (0.1%) misdetection was reduced from 32.2% to 23.3% (28% reduction). For low false acceptance rate (1%) misdetection was reduced from 28.9% to 21.1% (27% reduction).

Future work will focus on the following directions:

- Speeding up the DTW spotter during testing.
- Improving the speaker-dependent algorithm (DTW-spotter).
- Selecting templates from the training set without using any transcription, according to a DTW-matching criterion.

## 5. Acknowledgements

This work was partially supported by the KITE consortium of the Israeli Ministry of Industry and Commerce.

## 6. References

- [1] Sturim, D. E., Reynolds, D. A., Dunn, R. B., and Quatieri, T. F. "Speaker verification using Text-Constrained Gaussian Mixture Models", in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pp. 677-680, 2002.
- [2] Rabiner L. R. and Juang B.-H., *Fundamentals of Speech Recognition*, Prentice-Hall, 1993.
- [3] "Speech processing, transmission and quality aspects (stq); distributed speech recognition; front-end feature extraction algorithm; compression algorithms," ETSI Standard: ETSI-ES-201-108-v1.1.2, 2000, <http://www.etsi.org/stq>.
- [4] Reynolds, D. A., "Comparison of background normalization methods for text-independent speaker verification", in *Proc. Eurospeech*, pp.963-966, 1997.
- [5] Linguistic Data Consortium, SPIDRE documentation file, [http://www ldc.upenn.edu/Catalog/readme\\_files/spidre\\_readme.html](http://www ldc.upenn.edu/Catalog/readme_files/spidre_readme.html)
- [6] McLaughlin, J., Reynolds, D. A., and Gleason, T., "A study of computation speed-ups of the GMM-UBM speaker recognition system", in *Proc. Eurospeech*, pp.1215-1218, 1999.
- [7] "The NIST year 2002 speaker recognition evaluation plan", <http://www.nist.gov/speech/tests/spk/2002/doc/>.