TEXT-DEPENDENT SPEAKER-RECOGNITION USING ONE-PASS DYNAMIC PROGRAMMING ALGORITHM

V. Ramasubramanian *Amitava Das* *V. Praveen Kumar*

Siemens Corporate Technology – India
Siemens Information Systems Ltd., Bangalore – 560100, India
Email: {V.Ramasubramanian, Amitav.Das, V.Praveenkumar@siemens.com}

ABSTRACT

We propose a variable-text text-dependent speaker-recognition system based on the one-pass dynamic programming algorithm. The key feature of our proposed algorithm is its ability to use multiple templates for each of the words which form the "password" text. The use of multiple templates allows the proposed system to capture the idiosyncratic intra-speaker variability of a word, resulting in significant improvement in the performance. Our algorithm also uses inter-word silence templates to handle continuous speech input. Application of the proposed algorithm to two 100-speaker speaker-recognition systems, namely, closed-set speaker-identification (CSI) and speaker-verification (SV), delivers 100% speaker identification accuracy and a speaker-verification EER of 0.09%. The use of multiple templates (in comparison to a single template) enhances the CSI performance from 94% to 100% and the SV EER from 1.6% to 0.09%. Front-end noise suppression enables our systems to deliver robust performance in up to -10 dB car noise.

1. INTRODUCTION

Text-dependent speaker-recognition systems can be of two types: Fixed-text and variable-text. In either case, the password text can be a single word or a phrase of multiple words. For the fixed-text system, the password text is the same during training and testing. An isolated style dynamic-time warping (DTW) algorithm [1] [2] [7] is typically used to match the fixed-text test utterance with the fixed-text training template.

In contrast, a variable-text system uses a pre-defined vocabulary of words from which any 'variable' text can be composed and chosen by the user as the 'password', such as a Personal Identification Number (PIN). The system can also generate a password in a prompted-mode of operation. The variable-text system thus has the flexibility of allowing the composition of a variety of 'password' phrases. Even with a small vocabulary (such as digits 0 through 9), a large variety of passwords can be composed. Such flexibility of choosing the password as well as the ability to randomly generate a password by the system every time it is used are of paramount importance in voice-driven secure access control applications.

Despite the advantages of the variable-text operation, there were only a few results [3] [4] [7] reported in literature. Rosenberg et al. [3] used an isolated-style of DTW, where the input utterance (a string of words) is matched against the supposed 'password' text in the form of a concatenated sequence of the corresponding isolated word templates. Higgins et al. [4] used a connected word recognition algorithm [5] to perform either an open recognition of the input utterance or a forced alignment.

However, the shortcomings of both these approaches are as follows. The algorithm in [4] uses one averaged template per word per speaker for the forced alignment matching. This is clearly inadequate to handle intra-speaker variability (arising due to idiosyncratic pronunciation variations of the speaker or due to the inter-session variability over time). The isolated style DTW matching in [3] cannot handle multiple templates; moreover the isolated style of matching cannot also handle continuous input utterances, where there will be arbitrary inter-word pauses.

We propose a multiple-template based variable-text speaker recognition algorithm adopting the one-pass dynamic programming (DP) for the forced alignment matching. The one-pass DP algorithm was originally proposed for connected word recognition [6]. The use of multiple templates enhances the performance of our system by efficiently handling the intra-speaker variability. The use of inter-word silence templates allows users to freely speak the password in a continuous fashion as the system can now handle arbitrary inter-word silences gracefully while at the same time also allowing for inter-word co-articulations. Incidentally, due to this, the end-points of the isolated word templates are not so crucial in our system. As a result, a high degree of convenience is created for the user and system became more reliable as well.

Based on our proposed algorithm, two speaker-recognition systems for an 100 speaker task -- a closed-set speaker-identification system and a speaker-verification system -- were built, which deliver 100% speaker identification accuracy and a speaker-verification EER of 0.09%. The use of multiple templates enhances the CSI performance by nearly 6% over a single template system; the corresponding improvement in SV EER is from 1.6% to 0.09%. For real-life implementation, it is also important to make such system robust to background noise. We deployed a noise suppression technique at the front-end, which enables our systems to deliver robust performance in heavy noise conditions (up to -10 dB car noise).

2. PROPOSED ALGORITHM

The proposed variable-text speaker-recognition system based on the one-pass dynamic-programming (DP) matching algorithm and the corresponding system architecture is presented in Fig. 1. Here, each speaker has a set of templates for each word in the vocabulary. For example, for the word "nine", there are four templates, R_{91}, R_{92}, R_{93}, R_{94}. Given an input utterance, the feature extraction module converts the input speech utterance into a sequence of feature vectors. We used the mel-frequency-cepstral coefficients (MFCCs) as the feature vector. For example, when the password "915" is spoken by the user, a corresponding sequence of feature vectors is created and presented to the forced alignment module. At the same time, the corresponding concatenated set of multiple reference templates for "9", "1" and "5" along with the inter-word silence templates is also presented to the forced alignment module.

The one-pass DP algorithm matches the feature-vector sequence **O** against the multiple-template & inter-word silence based word-model sequence S_i for speaker-i. The resulting match score $D_i=D(O,Txt \mid S_i)$ is the optimal distance between the input utterance **O** and the word-templates of speaker S_i corresponding to the password text 'Txt'. This score is used in different ways in the two speaker-recognition systems as follows:

Speaker-identification (SI) system: The score D_i is computed for each of the N registered speaker in the system and the speaker with the lowest score is declared as the identified speaker.

Speaker-verification (SV) system: Here, the score D_i corresponds to the match between the input utterance and the templates of the 'claimed speaker' S_i. We perform a form of likelihood-ratio normalization on the one-pass DP score, D_i, by dividing it with the impostor score computed between the input utterance and a background speaker closest to the input utterance from among the remaining speaker set [4]. This normalized score is then compared to a threshold and the input speaker claim is accepted if the normalized score is less than the threshold and rejected otherwise. This is done for both target speakers and impostor speakers and the probabilities of false rejection and false acceptance for the given threshold are determined as defined in Sec. 3.3. This further yields the ROC curve for varying thresholds.

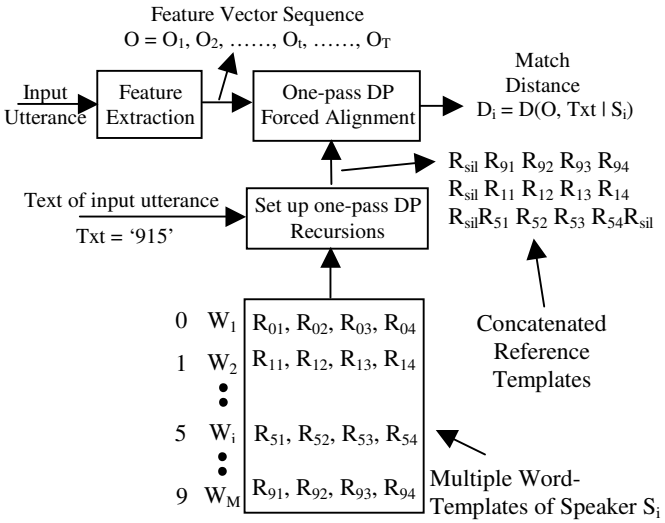

Fig.1 Proposed Variable-Text Speaker-recognition Algorithm based on One-pass DP Matching with Multiple Templates

2.1 One-pass DP algorithm with multiple templates and inter-word silence templates

Figure 2 illustrates the use of multiple templates in the proposed one-pass DP forced alignment between the input utterance (on the x-axis) and the word-templates (on the y-axis). The same example password of "915", as in Figure 1, is used. Even though multiple templates are being used for all the words, here for the sake of clarity, only the multiple templates of the word 1 are shown on the y-axis. From the best warping path obtained by the one-pass DP algorithm in this example, it is seen here that the template 2 of word 1 ($R_{1,2}$) had been chosen as the best matching template for that part (word '1') of the input utterance.

Figure 3 illustrates a typical matching by our proposed one-pass DP algorithm with templates for inter-word silences. In this example, it is assumed that the input utterance is the same ('915') as in Figure 1, but it is spoken with silence before 9, silence between 1 and 5 and after 5. There is no inter-word silence between 9 and 1, representing an inter-word co-articulation. The one-pass DP algorithm uses concatenated "multiple" templates of each word in the password '915' as in Fig. 2, but with a silence template between adjacent words (for the sake of clarity and also to emphasize the handling of inter-word silence, only one template per word is shown in Fig. 3). The one-pass DP recursions now allow for entry into any word either from a silence template or one of the multiple templates of the predecessor words. Figure 3 shows how the one-pass DP algorithm now correctly decodes the input utterance skipping the silence model between word 9 and 1. Other inter-word silences are mapped to the corresponding silence templates.

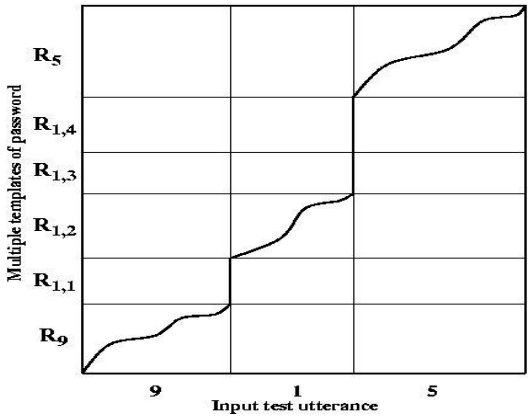

Fig. 2 One-pass DP matching between test utterance and multiple training templates corresponding to password text.

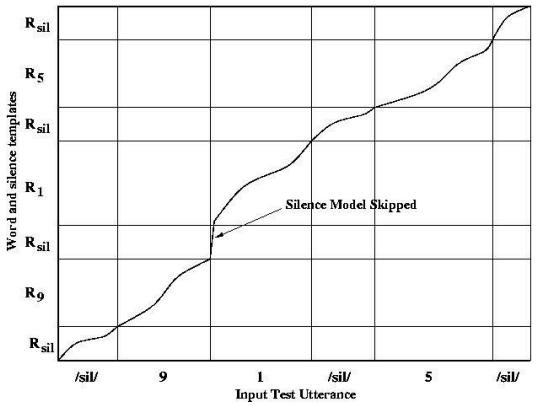

Fig. 3 One-pass DP matching with optional inter-word silences

We now state the dynamic program recursions, which are the heart of our one-pass DP algorithm, for the combined case of multiple templates and inter-word silence, illustrating how the warping paths (shown in Figs. 2 and 3) are realized jointly. The recursions for two specific parts, one for word-templates and the other for the inter-word silence templates, are presented next.

2.1.1 Word template recursions

Figure 4 shows the two main types of recursions, a) Within-word recursion and b) Across-word recursion for a general case of any word template, but in the context of the password-sequence '915'. The general equations for these two types of recursions are:

Within-word recursion

$$D(m,n,v) = d(m,n,v) + \min_{n-2<=j<=n} [\, D(m-1,j,v) \,]$$

Across-word recursion

$$D(m,1,v) = d(m,1,v) + \min \{ D(m-1,1,v), \min_{u \varepsilon Pred'(v)} D(m-1,N_u,u) \}$$

Here, D(m,n,v) is the minimum accumulated distortion by any path reaching the grid point defined as frame 'n' of word-template 'v' and frame 'm' of the input utterance.; d(m,n,v) is the local distance between the m-th frame of word-v template and n-th frame of the input utterance. The within-word recursion applies to all frames of word v template, which are not the starting frame (i.e., n>1). The across-word recursion applies to frame 1 of any word-v to account for a potential 'entry' into word v template from the last frame N_u of any of the other words {u} which are valid predecessors of word-v; i.e., Pred'(v) = {Silence template R_{sil}, Pred(v)}; these are the valid predecessors of any word v consisting of a silence template R_{sil} and the multiple templates Pred(v) of the word preceding the word v in the 'password' text; for instance, if the 'password' text is 915, and v=5, then Pred'(v=5) = {R_{sil}, R_{11}, R_{12}, R_{13}, R_{14}}; likewise, Pred'(v=1) = {R_{sil}, R_{91}, R_{92}, R_{93}, R_{94}}. This across-word recursion takes care of entry into any template of any word from a preceding silence template or from any template of any preceding word in the password text.

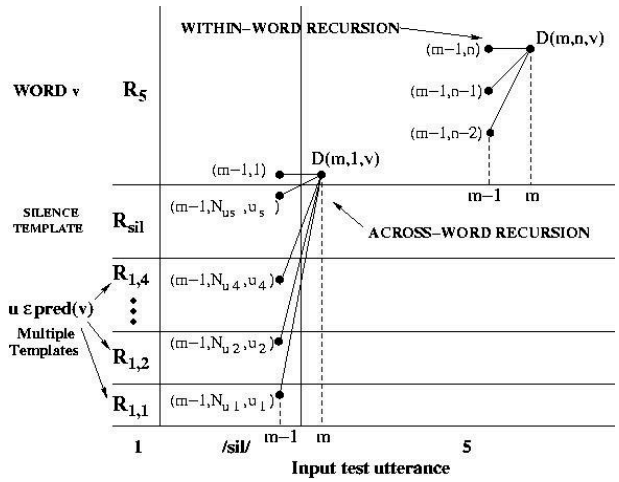

Fig. 4 One-pass DP recursions for optional inter-word silences

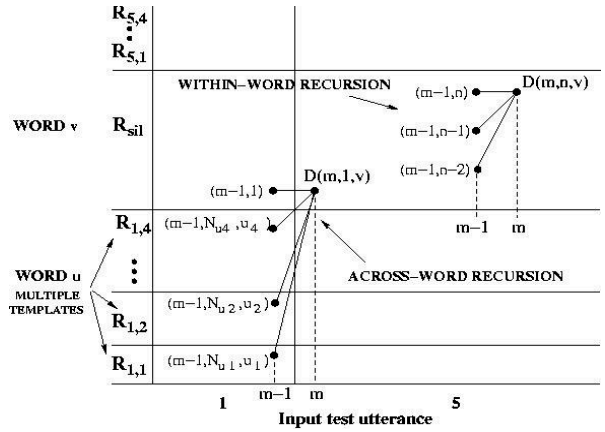

Fig. 5 One-pass DP recursions for multiple training templates

2.1.2 Silence template recursions

Figure 5 shows recursions for an inter-word silence template. This is illustrated for the transition from any of the 4 templates of word '1' to the silence template between words '1' and '5'. The within-word and across-word recursions in this case are:

Within-word recursion

$$D(m,n,v) = d(m,n,v) + \min_{n-2<=j<=n} [D(m-1,j,v)]$$

Across-word recursion

$$D(m,1,v) = d(m,1,v) + \min \{ D(m-1,1,v), \min_{u \varepsilon Pred(v)} D(m-1,N_u,u) \}$$

Here, all terms are same as in the recursions in Sec. 2.1.1 except the definition of Pred(v), where v is the inter-word silence template R_{sil} between two consecutive words in the password. Thus, Pred(v) is the set of the multiple templates of the preceding word in the 'password' text. For instance, if the 'password' text is 915, then Pred(v = R_{sil} between 1 and 5) = {R_{11}, R_{12}, R_{13}, R_{14}}, i.e., the 4 templates of word 1.

The above recursions together describe the one-pass DP recursion for using multiple templates and inter-word silence templates for forced alignment matching as required in the variable-text speaker-recognition. The score $D(T,N_r,r)$, where T is the last frame of the input utterance and word-r is the last silence template (with N_r as the last frame) yields the minimum accumulated distance D_i of the match between the input utterance and the 'password' text and is used as the score for that speaker-i whose word-templates were used. Beginning of section 2 already described how D_i is used for speaker-identification or speaker-verification.

3. PERFORMANCE OF PROPOSED ALGORITHM

3.1 Database

We have built a closed-set speaker-identification (CSI) system and a speaker-verification (SV) system using the proposed algorithm with 100 speakers from the TIDIGITS database. The TIDGITS database has a vocabulary of 11 words 'oh' and 0 to 9 with 77 continuously spoken digit string utterances per speaker of lengths 1, 2, 3, 4, 5 and 7 comprising of 22 utterances of length 1 and 11 utterances for each of the other lengths. We studied the proposed algorithm for test utterances of length 3, 4 and 5 digits pooled together. The training templates were excised from the 7-digit utterances, yielding up to 5 templates per word. We also studied the performance of the proposed system under noisy conditions, with 'car' noise being added digitally to the clean TIDIGITS database. The three noise-levels studied are clean, 0 dB and –10 dB SNRs. In all these cases, we have compared the performance of the system with and without noise-suppression techniques. The feature vectors used in the systems are MFCCs of dimension 12, obtained from an analysis frame size of 20ms and overlap of 10ms.

3.2 Closed-set speaker-identification system

Figure 6 illustrates the performance of the closed-set speaker-identification system as a function of the number of multiple templates used per word for various SNR conditions. Here, 33 test utterances (11 from each of lengths 3, 4 and 5 digits) per speaker per SNR condition, were used. Both training as well as test templates were subjected to noise-removal. The performance metric used here is the %SID accuracy, which is defined as the percentage of number of correctly identified test utterances from the total of 33x100 = 3300 trials for all speakers put together.

As seen in figure 6, the use of multiple templates clearly improves the CSI performance significantly at all SNR levels as compared to the single-template version. Particularly, the use of 5 templates yields 100% SID accuracy for SNR levels up to 0 dB.

To deliver high performance across all noisy condition, we have used noise-suppression at the front end during both training and testing. The impact of the noise-suppression is also seen in Fig. 6 for the -10 dB noise condition. Compared to the

performance when noise-suppression is not used (dotted blue line), the performance of the system with noise suppression (shown as dashed blue line) is about 10% better. When noise suppression is not used, even then the use of multiple templates is seen to deliver better performance than a single template version.

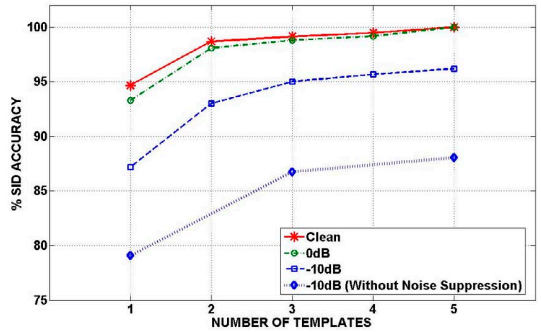

Fig. 6 Speaker-identification performance (% SID accuracy) for 1 to 5 training templates and for different SNR levels

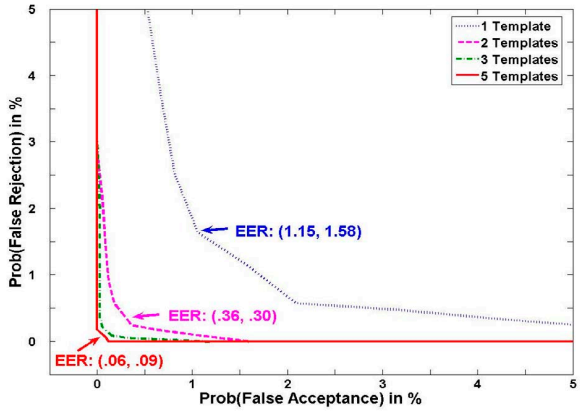

Fig. 7 ROC curves for the SV system for 1 to 5 training templates per word in CLEAN condition

3.3 Speaker-verification system

For the speaker-verification system, 33 test utterances (11 from each of lengths 3, 4 and 5 digits) were used as target speaker data, i.e., $N_{target} = 33$ per noise condition. For a target speaker, impostor data was generated from the non-target speakers (i.e., the other 99 speakers chosen randomly) to yield 33 test utterances (11 from each of lengths 3, 4 and 5 digits). Thus $N_{impostor}=33$. For all the 100 speakers, the total number of target and impostor trials are therefore $N_{target\text{-}trials} = 33x100 = 3300$ and $N_{impostor\text{-}trials} = 33x100 = 3300$.

If N_{fr} is the total number of times the system incorrectly rejects the claimed speaker, for a given threshold θ, the probability of false rejection is defined as:

$$P_{fr} = N_{fr} / N_{target\text{-}trials}$$

If N_{fa} is the total number of times the system incorrectly accepts an impostor, for a given threshold θ, as the corresponding claimed speaker, then the probability of false acceptance is:

$$P_{fa} = N_{fa} / N_{impostor\text{-}trials}$$

This yields a point (P_{fa}, P_{fr}) in the P_{fa}-P_{fr} plane for the given θ, and varying θ yields the ROC curve. The ROC curves were obtained for various number of training templates 1 to 5 and various SNR conditions. Figure 7 shows the ROC curve for the clean condition and for multiple templates. It can be clearly observed that the SV performance improves significantly with the use of multiple templates.

Table 1 presents the Equal-error-rate (EER) points (points on the ROC curve where $P_{fr} = P_{fa}$) for multiple number of templates and various SNR conditions. The results clearly show that a) the use of multiple templates indeed leads to higher SV performance and b) the front-end noise suppression makes our proposed algorithm quite robust to noise up to –10 dB, particularly with the use of multiple templates. Specifically, it can be noted that the SV system achieves a sub-0.1% EER which represents the best performance reported in literature so far for such a large set of speaker population (from which the impostors are drawn).

Table 1: EER for SV for various multiple templates and SNRs

# of temp	Clean		0 dB		-10 dB		All SNRs	
	P_{fr}	P_{fa}	P_{fr}	P_{fa}	P_{fr}	P_{fa}	P_{fr}	P_{fa}
1	1.58	1.15	1.85	1.73	5.42	4.33	3.09	2.24
2	0.30	0.36	0.45	0.33	2.27	2.15	1.21	0.74
3	0.15	0.09	0.18	0.24	1.52	1.58	0.40	0.76
4	0.09	0.09	0.15	0.21	1.45	0.94	0.66	0.29
5	0.09	0.06	0.09	0.12	0.94	0.79	0.58	0.22

4. CONCLUSIONS

We presented a noise-robust multiple template based one-pass DP algorithm for variable-text text-dependent speaker-recognition. The use of multiple templates allows efficient handling of the intra-speaker variability delivering significant performance improvement over single template version. The proposed algorithm also uses inter-word silence templates, enabling the speaker recognition system to handle continuous input utterances. The resulting 100-speaker speaker identification and speaker verification systems demonstrated high performance and robustness in noisy conditions up to –10 dB. The results reported here represent the best reported so far for text-dependent speaker-recognition for such large populations.

5. REFERENCES

[1] S. Furui, "Cepstral analysis technique for automatic speaker verification", IEEE Transactions on Acoustics, Speech and Signal Processing, 29:254-272, 1981.

[2] B. Yegnanarayana et al., "Combining evidence from source, suprasegmental and spectral features for a fixed-text speaker-verification system", IEEE Transactions on Speech and Audio Processing, 13(4):575-582, July 2005.

[3] A. E. Rosenberg, C. H. Lee, and S. Gokeen, "Connected word talker verification using whole word hidden Markov models" In Proc. ICASSP, pp. 381-384, 1991.

[4], A. Higgins, L. Bahler, and J. Porter, "Speaker verification using randomized phrase prompting", Digital Signal Processing, 1(2):89-106, 1991.

[5] J. S. Bridle, M.D. Brown and R.M.Chamberlain, "An algorithm for connected word recognition", Proc. ICASSP, Paris, 1982.

[6] H. Ney, "The use of one-stage dynamic programming algorithm for connected word recognition", IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP-32, no. 2, April 1984.

[7] V. Ramasubramanian and Amitava Das, "Text-dependent speaker-recognition – A survey and state-of-the-art", Tutorial at ICASSP-2006, Toulouse, France, May 2006, Accepted.