ROBUST RECOGNITION USING MULTIPLE UTTERANCES

Yoram Meron *Keikichi Hirose*

Dept. of Information and Communication Engineering, Faculty of Engineering
The University of Tokyo, Hongo 7-3-1, Bunkyo-ku, Tokyo 113-8656, Japan
(meron,hirose)@gavo.t.u-tokyo.ac.jp

ABSTRACT

Increasing the reliability of the results of automatic speech recognition systems is an important research and development issue. Although recent systems have been shown to achieve quite high recognition results for limited tasks, this may not be good enough for some applications, where some bits of information are critical, and have to be recognized correctly. In this paper, we suggest a method for the improvement of the robustness of speech recognition, using a speaker independent HMM, of either a word, phrase, or a full sentence, by taking advantage of repeated utterances of the same content. The method can be applied to most configurations of HMM based recognizers, and does not require additional model training. Recognition experiments showed improvement in recognition accuracy, under quiet and noisy conditions.

1. INTRODUCTION

In conventional recognition systems, when recognition fails, or when uncertainty regarding the correctness of the recognition arises, the system may ask the user to repeat the utterance. The system would then perform the same recognition process on the new utterance. Typically, the system would ignore the first utterance, and use only the result from the repeated utterance (possibly asking for yet another utterance if it suspects the recognition failed again). By doing this, the system obviously does not take advantage of the information available in the discarded utterance.

In the more sophisticated case, systems try to use information from both utterances, by incorporating some decision strategy. The simplest strategy is selecting the recognition result (i.e. phone sequence) which received the better recognition score. A further refinement of this strategy is to run a forced alignment on both utterances, over the two[1] recognition results, and choose the sequence which got the best combined score for the two utterances.

All of the above methods have several disadvantages. First, they assume one of the obtained recognition results was the correct result (since one of them will be selected as the final result). Second, they look only at the global behavior of the recognizer (phone sequence and recognition score), and ignore what happens locally (the frame-level path found by the recognition decoder).

The method we suggest does not ignore this local behavior of the recognizer over the utterances. Instead of running the recognizer over the utterances one by one, we force the decoder to run *"simultaneously"* over all the (synchronized) utterances at once, thus forcing the recognizer to select the (frame level) recognition path which best matches *all* of the utterances.

2. PROPOSED METHOD

The suggested recognition method works in the following way: The speaker utters the same input two or more times. For each utterance $(U_1, U_2, ...)$ A DTW algorithm is used to produce a time alignment $(T_1, T_2, ...)$ to a common time base (e.g. align all utterances to U_1).

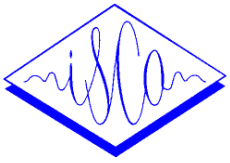
The recognizer, a speaker independent HMM, is modified in order to be able to run on several utterances simultaneously. In the original HMM algorithm, the dynamic programming algorithm calculates a *local distance* (likelihood) grid $S(i,j)$ as $dist(V(i), H(j))$ - the computed distance between an HMM state $H(j)$ and a feature vector $V(i)$, extracted from one acoustic frame $A(i)$ (this is typically calculated as the distance between a Gaussian mixture and a feature vector).

In the modified version, $S(i,j)$ is calculated as an average of the distance between the HMM state $H(j)$ to several feature vectors- $V_1(k_1), V_2(k_2)...$ - one from each utterance, such that $T_1(k_1) = T_2(k_2) = ... = i$ (i.e. vectors which were aligned to time i in the common time-base). This is illustrated in figure 1.

The method relies on the fact that the result of the DTW alignment, for the same utterance by the same speaker, is more robust than result of a speaker independent HMM. Thus, the method forces the HMM to traverse all of the utterances through the "same" path, and helps the HMM avoid falling into wrong local minima for each utterance separately.

The suggested method can be regarded as an add-on to the recognizer, which can be applied to most HMM configurations (various kinds of feature vector types, search restrictions, feature scoring, probability models, etc.). Specifically, the acoustic models, used by the rec-

[1] The methods described in this paper can be directly applied to more than two utterances.



6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

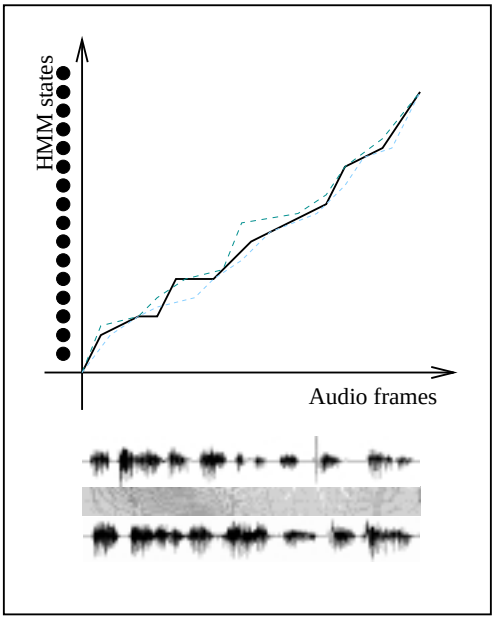


Figure 1: Proposed HMM decoding process - forcing the decoder to find one path (dark line) for two aligned utterances, avoids falling into (different) local minima when running the decoding separately for each utterance (light lines).

ognizer, do not have to be re-trained - existing models, trained by existing training methods, can be used.

2.1. Cepstrum averaging

A simpler approach for taking advantage of multiple utterances was also tested. In this approach, the repeated utterances are also time aligned, but the time alignment is used for creating one "averaged utterance" - for each frame, an average of the feature vector's coefficients over all the time aligned utterances are calculated. The resulting feature vector sequence is then given to the decoder, which treats it in the same way it does features sequences extracted from natural utterances.

The two approaches differ in that the first performs the averaging over the *spectral distance space*, while the second performs the averaging over the *cepstral coefficients space*, which, theoretically, is less justifiable.

3. IMPLEMENTATION NOTES

3.1. Base system

The base system used for this work was the HMM based JULIAN speech recognition system (version 2.2) [1], developed at the university of Kyoto. For models, we used two sets of recognition models, both for continuous speech, speaker independent, gender specific (male) models for Japanese, which are supplied with the system. One set included 43 monophone models, using 3 states per model, and 16 Gaussian mixtures per state. The other set includes 1000 triphone states, with 4 mixtures per state. Feature vectors are composed of 25 components computed each 10 msec, consisting of mel cepstrum, delta mel cepstrum, and delta energy coefficients.

3.2. System modifications

The base system was modified to perform the proposed method (the experiments described here were all performed for the special case of two or three utterances, but, as mentioned before, the method can easily be extended for any number of utterances).

First, a standard DTW algorithm [2] was implemented. Next, the feature vectors of a repetition of the utterance are DTW aligned with the first utterance, and a time aligned version is stored for the following processing. Last, a simple modification was applied to the calculation of the *local score* in the HMM decoder:

$$\hat{S}(i,j) = \frac{1}{N}\sum_{k=1}^{N} dist(V_k(T_k(i)), H(j))$$

For the averaged utterance method, an option to average 2 or more time aligned feature sequences was added to the original program.

4. TEST SPEECH DATABASE

In order to test the performance of the proposed methods, a test speech database was collected. The test database consisted of recordings of 6 (male) native Japanese speakers (non-professional speakers). The test data consisted of 2 parts - sequences of digits, and sequences of (Roman) letters. Each part was made of 3 repetitions of 10 sequences of 10 digits (or 10 letters), with the order of appearance of the sequences changed. In total, each of the sequences was uttered 3 times by each of the 6 speakers.

The speakers were instructed to speak naturally, as if reading a telephone number, or spelling out a word. No restrictions on digit grouping or any specific reading rhythm were enforced.

The recording was done in an acoustically isolated room, using a DAT, and then sampled at 16kHz, 16 bit, using DATlink.

5. RECOGNITION EXPERIMENT

A recognition experiment was run on the test data, testing the recognition accuracy rate ($\frac{N-(d+i+s)}{N}$) and correct rate ($\frac{N-(d+s)}{N}$), where N is the number of digits / letters in the sequence, and d, i, s are the number of deletions, insertions and substitutions, respectively (using the "best case" match between the correct and the recognized sequences). These rates were tested separately for the digit sequences and the letter sequences, using either the monophone or triphone models, for each of the recognition methods:

- **Single** - running the original system on one utterance at a time

- **Best of 2** - running the original system once on each of a given pair of utterances, selecting the utterance which got a better recognition score (overall log likelihood), and taking the recognized sequence of that utterance to be the recognition result.
- **Best of 3** - the same, for 3 utterances
- **Multi 2** - running the proposed method (simultaneous decoding on aligned sequences) on two utterances
- **Multi 3** - the same for 3 utterances
- **Aligned 2** - averaging the cepstral vectors over an aligned pair of utterances
- **Aligned 3** - the same for 3 utterances

For the digits, the vocabulary size was 20 words, including several phonetic variations of the digit names (e.g. for the digit 7: "nana", "shichi", "shchi", "shich") and silence. No language model, or restrictions on the grammar were used. Especially, the decoder was not restricted to output sequences of 10 digits.

For the spelling, The vocabulary size was 45, again including several pronunciation variations of English letter names by native Japanese speakers.

Tables 1, 2, 3, 4 show the recognition results for digits vs. letters, and monophone vs. triphone models. These results are also summarized in figure 2.

Manual inspection showed that the DTW did not produce gross alignment errors: audio files, with the automatically time aligned speech imposed over the original speech were created. Informal listening to these files did not detect any noticeable alignment errors.

Condition	Accuracy rate	Correct rate
Single	88.8%	93.1%
Best of 2	88.9%	92.9%
Best of 3	90.5%	93.5%
Multi 2	94.9%	96.3%
Align 2	94.6%	96.1%
Multi 3	98.0%	98.0%
Align 3	97.7%	98.0%

Table 1: Recognition performance, digit sequences, monophone models

Condition	Accuracy rate	Correct rate
Single	89.1%	93.0%
Best of 2	89.9%	93.3%
Best of 3	90.0%	94.5%
Multi 2	94.6%	96.4%
Align 2	92.8%	95.1%
Multi 3	95.0%	96.5%
Align 3	94.5%	96.0%

Table 2: Recognition performance, digit sequences, triphone models

The result show several consistent trends:

Condition	Accuracy rate	Correct rate
Single	26.9%	74.7%
Best of 2	25.1%	74.2%
Best of 3	28.5%	74.0%
Multi 2	42.0%	75.7%
Align 2	38.5%	75.9%
Multi 3	43.0%	74.5%
Align 3	43.5%	75.5%

Table 3: Recognition performance, letter sequences, monophone models

Condition	Accuracy rate	Correct rate
Single	26.9%	75.0%
Best of 2	25.1%	75.3%
Best of 3	27.0%	75.0%
Multi 2	31.4%	75.1%
Align 2	35.1%	74.5%
Multi 3	50.5%	76.0%
Align 3	31.5%	69.5%

Table 4: Recognition performance, letter sequences, triphone models

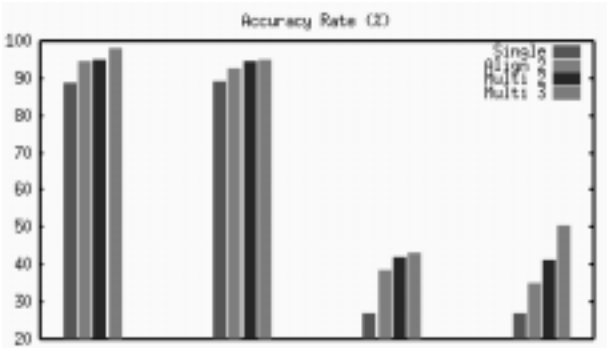


Figure 2: Results of recognition test for the Single/Align2/Multi2/Multi3 methods, run for (from left to right): digits (monophone, triphones), letters (monophone, triphones)

- As expected, the letter recognition suffers from a much higher error rate - both because of larger vocabulary, and the occurrence of easily confuseable words.
- Comparing the single/best2/best3 methods, there was only little improvement (if at all) for using more than one utterance.
- Comparing the single/align2/align3 methods, there was improvement by using more utterances.
- Comparing the single/multi2/multi3 methods, there was clear improvement by using more utterances.
- Comparing the multi2-align2 and multi3-align3 methods, the multi2/3 consistently performed better .
- For the letter recognition, most of the improvement in recognition went to reduce the number of insertions (the improvement in the correct rate, which does not depend on insertions, was much smaller than the improvement in the accuracy rate).

- These trends did not depend on the use of monophone/triphone models.

6. NOISY SPEECH

The same experiment was conducted for noisy speech, to test the applicability of the proposed method to increasing the robustness of recognition for noisy speech.

White Gaussian noise, with variable SNR values, was added to the original speech files, and the recognition was run again. The SNR levels used were 20[dB], 10[dB], 5[dB], and 0[dB].

The recognition results (using the digit sequences and the monophone models) are shown in tables 5, 6, 7, 8, and are summarized in figure 3.

Condition	Accuracy rate	Correct rate
single	87.6%	93.3%
best of 2	89.0%	93.8%
best of 3	92.0%	95.0%
multi 2	91.6%	95.1%
align 2	89.9%	94.1%
multi 3	94.5%	96.5%
align 3	95.0%	97.0%

Table 5: Recognition performance, digits, monophones, SNR=20[dB]

Condition	Accuracy rate	Correct rate
single	75.7%	85.2%
best of 2	78.2%	86.2%
best of 3	79.5%	86.0%
multi 2	76.2%	85.1%
align 2	74.4%	83.4%
multi 3	80.0%	87.0%
align 3	75.5%	84.0%

Table 6: Recognition performance, digits, monophones, SNR=10[dB]

Condition	Accuracy rate	Correct rate
single	47.6%	62.1%
best of 2	47.2%	61.7%
best of 3	45.5%	60.5%
multi 2	50.3%	61.9%
align 2	44.0%	57.5%
multi 3	52.5%	62.0%
align 3	42.5%	54.5%

Table 7: Recognition performance, digits, monophones, SNR=5[dB]

The results show that the proposed method continues to increase recognition rates for lower SNR levels as well, although the absolute rise tends to decrease.

Manual inspection showed that the DTW continued to produce correct alignment results for the noisy speech.

Condition	Accuracy rate	Correct rate
single	22.4%	36.4%
best of 2	21.9%	35.3%
best of 3	22.0%	31.5%
multi 2	22.9%	33.6%
align 2	21.1%	31.9%
multi 3	26.0%	33.0%
align 3	18.5%	26.5%

Table 8: Recognition performance, digits, monophones, SNR=0[dB]

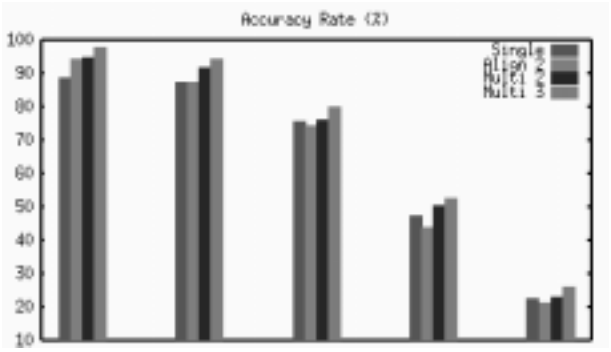


Figure 3: Results of recognition test for the Single/Align2/Multi2/Multi3 methods, using monophone models on digit sequences, run for different SNRs (from left to right): quiet, 20[dB], 10[dB], 5[dB], 0[dB]

7. CONCLUSION

A method to improve speech recognition accuracy, using multiple utterances of the same sentence, was introduced. The method forces an HMM decoder to run over several time aligned utterances, forcing it to traverse all the utterances "in the same way", thus avoiding the selection of different local minima for the different utterances.

Recognition experiments verified that this method can improve recognition rates, for several test configurations. Testing recognition for different levels of added noise showed the method can be used for noisy speech as well.

Further work is planned, applying this idea to improving the accuracy of automatic segmentation (using repeated utterances).

8. REFERENCES

1. T. Kawahara et al., "Shareable software repository for Japanese large vocabulary continuous speech recognition", Proc. ICSLP'98, pp. 3257-3260
2. L. Rabiner, B.H. Juang, "Fundamentals of speech recognition", Prentice Hall signal processing series, 1993