

AN ALGORITHM FOR INCREASING SPEED IN DYNAMIC TIME WARPING

J.A.S. Angus*, M.T. Whitaker*.

ABSTRACT

One of the more successful algorithms used in speech pattern matching is dynamic time warping, however, when applied to large vocabulary systems or when used for connected speech recognition, it uses a considerable amount of processing time. The most time consuming part of the algorithm is the calculation of the distance between two frames of parameters. If the two frames are highly dissimilar then the exact distance between them is not very important, and the time spent calculating it is wasted. This paper suggests the use of a quick distance measure which produces a fixed distance for obviously dissimilar frames, while passing the remainder on to a more detailed distance measure. The quick measure described is capable of providing a factor of four improvement in the speed of the dynamic time warping algorithm.

INTRODUCTION

Dynamic time warping has been used with considerable success in speech recognisers to find the optimum time alignment between two utterances of a word. In its simplest form, the algorithm requires the calculation of a square array of distances, representing the spectral correlation between the two templates (fig 1b). The size of this array is determined by the lengths of the two utterances. From this an array representing the minimum culmulative distance to each point can be calculated, using the equation

$$\text{mcd}[i,j] = \min(\text{mcd}[i-1,j], \text{mcd}[i-1,j-1], \text{mcd}[i,j-1]) + \text{dist}[i,j]$$

where mcd is the minimum culmulative distance array, dist is the spectral correlation array, and min returns the minimum value of its parameters. This can be combined with the previous step to save time and space, as each distance is only used once. By back-tracking from the top right to bottom left element, along the path of minimum distance, the optimum time alignment of the two utterances can be determined (fig 1c). A horizontal step on this path represents compression of the horizontal utterance at that point, a vertical step represents expansion of the horizontal utterance.

This algorithm requires a large amount of processing, mainly in the calculation of the spectral correlation array, so various techniques have been used to reduce the number of calculations required. The most common technique is to limit the time alignment path to a certain area of the array, some typical path constraints being shown in figure 2. Obviously the distances outside these constraints need not be calculated as they will not be used. This technique is often favoured as it prevents too gross a distortion of the utterances, however, the same path constraint may not be suitable for all the words in the vocabulary. It is also

*Department of Electronics, University of York, York, England.

difficult to apply this sort of constraint when using dynamic time warping for connected word recognition (ref 1).

An alternative technique for reducing processing time is the use of vector quantisation on the input parameter frames. A set of likely parameter frames (vectors) is stored and each input frame is compared with these and the closest match is chosen. The chosen vector is then used to represent that frame during the rest of the processing. The distance between all possible pairs of vectors can be calculated and stored before the time warping is performed, thus saving considerable time in the algorithm. The disadvantage of this method is that it reduces accuracy, due to the quantisation process, and that it requires additional training, in order to select the set of likely vectors.

For points in the spectral correlation array where the distance is high the actual value of the distance is fairly unimportant, as the best path is unlikely to go through many of these points. If these points could be easily located, and a fixed distance assigned to them, then considerable processing time could be saved, providing that the location procedure takes much less time than the ordinary distance measure. A scheme of this sort has been proposed by Zue (ref 2), using a broad phonetic classification scheme to select frames for more detailed matching. The problem with this sort of scheme is that the initial selection process must be extremely reliable. This is difficult to achieve with a phonetic based decision scheme, particularly during phoneme transitions. Our proposal uses a simple measurement of spectral shape, which, being related to our detailed distance measure, should prove fairly reliable in selecting similar frames.

IMPLEMENTATION

Our parameter frames consist of 30 bands representing the log power spectrum. The bands are spaced linearly up to 2.4kHz and in gradually wider bands from there up to 6kHz. The normal distance measure used is

$$\text{dist} = \sum (\text{frame1.band}[i] - \text{frame2.band}[i])^2 \quad (i = 1..30)$$

This measurement is scaled and limited so that the maximum distance is 255.

For the quick distance measure, the frequency space is split into five bands (boundaries at 0.5, 1.1, 1.9, 3.1kHz). The energy in each of these bands is compared to an energy threshold, and if above the threshold, the appropriate bit in a 5 element bitset is set. This yields a number between 0 and 31 which gives a rough representation of the spectral shape of the frame.

The problem with this method is that if the energy in any of the five bands is close to the energy threshold then similar frames may easily give rise to different quick measures. To solve this problem the reference templates should hold the quick measure in a 32 bit bitset. By training the recogniser with several utterances of each word, all the likely variations in the quick measure can be included in the set. The disadvantage of this solution is of course that more frames are likely to be selected for matching with the normal distance measure, but with

careful choice of energy thresholds and band spacing a significant improvement in speed should still be obtainable.

The time warping routine becomes

```
FOR i:=1 TO inputlength DO
  FOR j:=1 TO referencelength DO
    mcd[i,j]:=min(mcd[i-1,j],mcd[i-1,j-1],mcd[i,j-1]);
    IF input[i].quickmeasure IN reference[j].quickmeasure
      THEN mcd[i,j]:=mcd[i,j] + distance(input[i],reference[j])
      ELSE mcd[i,j]:=mcd[i,j] + 255
    ENDIF
  ENDFOR
ENDFOR
```

RESULTS

The above algorithm has been tested in a simple recognition system using the digits vocabulary. Only a single training utterance was used for each word in the vocabulary, hence the problems outlined in the previous section are to be expected. It was found that the word "five" was commonly misrecognised, but that otherwise the performance of the recogniser was as good as with the quick measure disabled. The average recognition delay with the quick measure was 1.2 seconds, compared to 5.5 seconds without the quick measure, a factor of 4.6 improvement.

Spectrograms of two utterances of the word "six" are shown in figure 1, along with the spectral correlation between the two, the best warp path, and the masking pattern produced by the quick distance measure. These two utterances highlight the problem caused by energies near the threshold - the small amount of low frequency energy present in the final /s/ of the first utterance has caused incorrect masking of the first half of the phoneme.

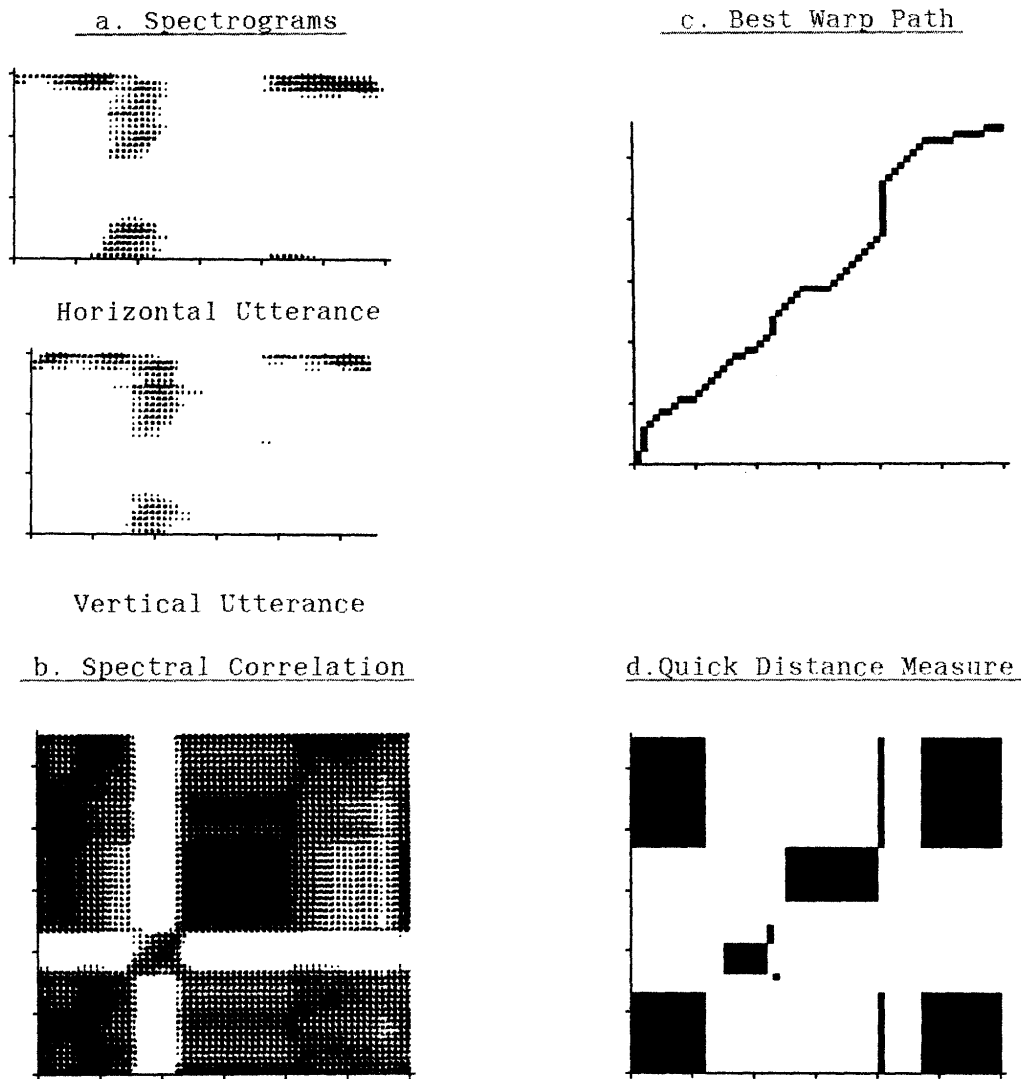
CONCLUSION

This paper has proposed a method of speeding up dynamic time warping without putting constraints on the allowable path or using vector quantisation. The improvement in speed is roughly comparable to that gained by applying a fairly restrictive path constraint. Initial testing shows that for most words recognition accuracy is not adversely affected, and it is expected that with an improved training procedure, the remaining inaccuracies will be eliminated.

REFERENCES

1. J.S. Bridle, M.D. Brown, R.M. Chamberlain, Radio and Electronic Engineer, Vol 53, pp167-175 (1983).
2. V.W. Zue, Proc. IEEE, Vol 73, pp1602-1615 (1985).

Figure 1. Comparison of Two Utterances of the Word "Six".



Black level represents closeness of match

Figure 2. Two Possible Path Constraints.

