BRITISH ACOUSTICAL SOCIETY

"SPRING MEETING" at Chelsea College, London, S.W.3 on
Wednesday 25th April / Friday 27th April, 1973.

SPEECH AND HEARING: Session 'C': Speech Properties and Recognition.

Paper No:

73SHC3

An Efficient Elastic-Template Method for Detecting Given Words in Running Speech

J S Bridle

Joint Speech Research Unit

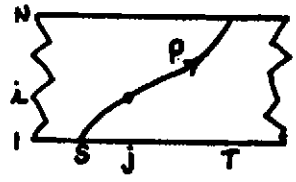
1. Introduction In many applications of automatic speech recognition, such as voice input to computers, it will be necessary to decode a spoken command to produce a sequence of words which 'makes sense'.

In general, two utterances of the same word have different time-structures because of differences in pronunciation: some parts of the word become longer, others shorter, to varying extents. One approach to automatic speech recognition (the segmentation approach) attempts to identify phonemes as they occur, and thus reduce speech to a string of symbols similar to that which a phonetician might use to represent the utterance. Unfortunately there is no simple relationship between acoustic and phonetic descriptions of a speech signal.

An alternative approach is a template method in which we first use a measurement process to represent the speech signal as a small set of slowly varying functions of time. Words in the vocabulary are stored as 'templates' derived from actual spoken examples of the words. Typically, a template might be 50 regularly spaced samples of six measures. We assume that the same, or similar, words in the input speech will produce signals which are similar to the template, except that the time scales will be different. This assumption of similarity may of course turn out be as wrong as the acoustic-phonetic one above, but at this stage the possibilities for suitable measurement methods have not been fully explored.

In what follows we simplify the problem of recognising a sequence of words by considering the identification of a single word, represented by one template, but we retain the requirement that the word be picked out of a complete sentence, without prior segmentation into word units.

2. The non-linear time-distortion problem Before we can measure how similar to the template a portion of the input is, corresponding parts must be aligned in time. On a time-time diagram with time for the template drawn vertically and the input horizontally, any particular way of arranging the samples of the template against the samples of the input can be represented as a time-registration 'path', P, from the bottom to the top.



At any point (i, j) we can find the instantaneous similarity, A(i, j) between the ith time-sample of the template and the jth time-sample of the input. The method of computing A should depend on the nature of the signals resulting from analysis of the speech.

A total fit function, TF(P), is a measure of the overall similarity between the template and the portion (S-T) of the input between the ends of

such a path, evaluated along the path. TF should include the instantaneous similarities, the amount of time-scale distortion and the length of the path. We could say that an example of the template word occurs in the input if there is a complete path for which TF is greater than some threshold.

The problem is to find where such paths occur, without trying all possible paths. (Even on a 10 by 10 diagram, the number of paths from corner to corner of the type described in section 5 is more than one million).

3. Two methods One approach is to use a path-growing heuristic [1] which attempts to find good time-registration paths by examining the instantaneous similarity, A, in various directions from the growing tip of the path. Warren [2] and the author's unpublished work indicate that developments of such methods can give good results some of the time, but that wrong decisions can cause failure, even with two similar patterns.

A better approach is to use the principles of Dynamic Programming [3] to consider all optimum paths. It is only possible to do this properly if the fit function has a rather special property: that the best way, $\hat{P}(i, j)$, of getting to a point (i, j) should be independent of what happens beyond (i, j). In this case there exists a 'partial function', F(P(i, j)) which defines $\hat{P}(i, j)$ as the partial path to (i, j) which maximises F(P(i, j)).

$$\text{ie} \quad \hat{F}(i, j) = F(\hat{P}(i, j)) = \underset{P}{\text{Max}} \left\{ F(P(i, j)) \right\}$$

F can be defined incrementely:

$$F(P(i-\alpha, j-\beta), (i, j)) = G(F(P(i-\alpha, j-\beta)), A(i, j), \alpha, \beta)$$

where $(i-\alpha, j-\beta)$ is the point on the path just before (i, j).

The crucial step is to recognise that $\hat{F}(i, j)$, (and therefore $\hat{P}(i, j)$) can also be defined (and computed) by a simple local operation:

$$\hat{F}(i, j) = \underset{\alpha, \beta}{\text{Max}} \left\{ G(\hat{F}(i-\alpha, j-\beta), A(i, j), \alpha, \beta) \right\}$$

The sequence of values $TF(\hat{P}(N, T))$ can be thought of as the output of a time-flexible generalisation of a matched filter, and a detection threshold could be applied to decide where examples of the template word occur.

Now we must consider the all-important requirement on TF: the ability to recognise the first part of an optimum path without considering the rest of it. If the beginning and end of a candidate portion of input speech could be identified first, we could use a path-engtth normalisation factor in TF which is independent of details of the path [4], but if we want to consider all paths on a strip, then at an intermediate point we cannot know how long the complete path will be. Therefore we cannot decide which of two paths up to this point is better if one has a high score so far but is long, and the other has a low score but is short.

It appears that it is too much to ask a method like this to detect the words as well as measure similarity. If we could detect, by other means, just one reliable acoustic event near the middle of the template word, then we could evaluate TF for the best path through this fixed point (always working towards the fixed point). The alternative, described below, is to concentrate on the detection aspect.

4. The new method F is defined for any partial path to (i, j), incrementy, as

$$F(P(i-\alpha, j-\beta), (i, j)) = (1-\gamma).\, F(P(i-\alpha, j-\beta) + \gamma.\, H(A(i, j), \alpha, \beta)$$

with starting condition F(1, j) = A(1, j)

Thus F is a sum of recent values of instantaneous similarity along the path,

with past values given exponentially decreasing weight, according to the value of γ. H applies an extra factor to A, according to the local time-distortion which is a function of the increments α and β.

F is a measure of the 'local similarity' at any point on a time registration path. We now say that a portion of the input pattern between points S and T is significantly similar to the template if there is a path from (1, S) to (N, T) for which F is above a threshold at every point on the path.

This criterion, which is superficially similar to the one for detection using total fit, in section 3, is very easy to implement. Paths are considered 'broken' as soon as F falls below the threshold, and $\hat{F}$ is defined in terms of unbroken paths. This results in a considerable saving of computation, because there will be large areas of the time-time diagram which cannot be reached by unbroken paths, and the instantaneous similarity, A, need not be evaluated in such regions.

When we find that a best unbroken path has reached the end of the template, then we have a section of the input which is locally similar to the template throughout its length, with due allowance for distortion of time scales.

5. Details of present implementation A(i, j), the instantaneous similarity between the ith time-sample of the template and jth sample of the input, is computed from the Euclidean distance, d, between the two points in measurement space.

$$A(d) = 1/(1 + d^2/\delta^2)$$

A is a bell-shaped function of d, with a maximum value of 1 when d = 0, and a value of 0.5 when d = δ.

δ is made variable over the template, to allow for greater variation in some parts of the word, particularly at the beginning and the end.

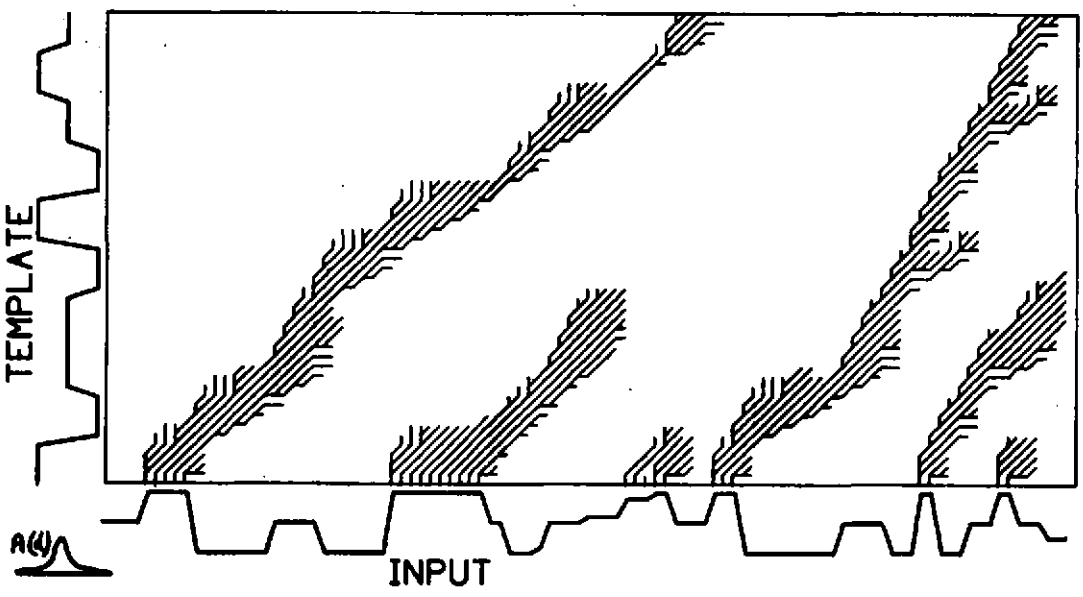
Paths consist of horizontal, vertical and diagonal segments, as in [4]

ie $(\alpha, \beta) = \{(1, 0), (0, 1), (1, 1)\}$

In the expression for F, A is attenuated by a factor k if a horizontal or vertical segment, corresponding to time distortion, is used.

When the point (i, j) is being considered, a one dimensional array of length N holds

$\hat{F}(1, j), \hat{F}(2, j), \ldots \hat{F}(i-1, j), \hat{F}(i-1, j-1), \hat{F}(i, j-1), \ldots \hat{F}(N-1, j-1)$



and the newly - calculated F(i, j) replaces F(i-1, j-1).

The figure illustrates the pattern of best unbroken paths produced in response to input sequences which are distorted in amplitude and time with respect to the template. The template and the input are artificial one-dimensional functions of time, chosen to illustrate the action of the process. δ is constant, and the resulting A(d) is illustrated on the same scale as the template. k = 0, so there is a high penalty for time-distortion. γ = 0.3 and threshold = 0.35. Note that the paths illustrated are optimum in terms of local fit only. They would not be best for an overall similarity measure.

6. Conclusions The method described will detect, in continuous functions of time, all portions which are locally similar to a template function throughout their length, with allowance for distortions of the time scale. The criterion for local similarity is a fairly simple one, and the smoothing time-constant and the sensitivity to distortions of amplitude and time can be adjusted separately.

Implementation is straightforward, and the amount of computation needed for each input sample is low, although the use of information from other sources (eg semantic information, or detection of particular acoustic events) may reduce this still further.

The method does not give a means of computing the overall similarity between the template and the isolated portion of the input. This could be an important disadvantage where a choice has to be made between two or more possible words.

It would seem difficult to incorporate into a method like this much of our detailed knowledge of the nature of the speech signal, or possible knowledge of the structure of the utterance which is being analysed.

Any application to automatic speech recognition depends on finding suitable methods of preliminary analysis of speech, and the associated instantaneous similarity function [5].

Since there is often more than one way of saying a word, it may be necessary to use more than one template per word [6], or to have a set of templates for each speaker.

7. References

1. Ellis, J. H., Electronics Letters, Vol 5, 335-6, July 1969.
2. Warren, J. H., 'A Dynamic Pattern Matching Algorithm with Applications to Automatic Speech Recognition', in Machine Perception of Patterns and Pictures, Institute of Physics, London, 1972.
3. Bellman, R. E., Dynamic Programming, Princeton Univ. Press, 1957
4. Velichko, V. M. and Zagoruko, N. G., 'Automatic Recognition of 200 Words', Int. J. Man-Machine Studies, Vol 2, 223-234, 1970.
5. Nakano, V. et al,'Evaluation of Various Parameters in Spoken Digits Recognition', Proc IEEE-AFCRL Conference on Speech Communication and Processing, 101-104, 1972.
6. Shearme, J. N. and Leach, P. F., 'Some Experiments with a Simple Word Recognition System', IEEE Trans Audio and Electroacoustics, Vol AU-16, No 2, 256-261, June 1968.