TIME WARPS, STRING EDITS, AND MACROMOLECULES: THE THEORY AND PRACTICE OF SEQUENCE COMPARISON

Edited by

DAVID SANKOFF
UNIVERSITY OF MONTREAL
MONTREAL, QUEBEC, CANADA

and

JOSEPH B. KRUSKAL
BELL LABORATORIES
MURRAY HILL, NEW JERSEY

1983

ADDISON-WESLEY PUBLISHING COMPANY, INC.
Advanced Book Program
Reading, Massachusetts
London • Amsterdam • Don Mills, Ontario • Sydney • Tokyo

CHAPTER FOUR

THE SYMMETRIC TIME-WARPING PROBLEM: FROM CONTINUOUS TO DISCRETE

Joseph B. Kruskal and Mark Liberman

1. INTRODUCTION

A trajectory, as illustrated in Fig. 1, means a continuous function of time in multidimensional space, i.e., a time-labelled curve in multidimensional space. Time-warping, as illustrated in Fig. 2, refers to comparison of trajectories, or to comparison of sequences derived from them by time-sampling, when each trajectory is subject not only to alteration by the usual additive random error but also to variations in speed from one portion to another. (In some applications, it is necessary to permit other differences between the trajectories as well, such as deletion and insertion, but we touch on that only lightly.) Such variation in speed appears concretely as compression and expansion with respect to the time axis, and will be referred to as compression–expansion. The chief purpose of time-warping is to deal with such variation. The chief application has been to speech processing, where compression–expansion is of major importance.

Time-warping is used in at least three ways. One is to discover the pattern of compression–expansion that connects two sequences. Another is to measure how different two sequences are in a way that is not sensitive to compression–expansion but is sensitive to other differences. A third use is in forming the weighted "average" of two sequences.

Time-warping of sequences is very similar in form and methodology to the comparison of "naturally discrete" sequences discussed elsewhere in this volume, such as the macromolecules of molecular biology and the character

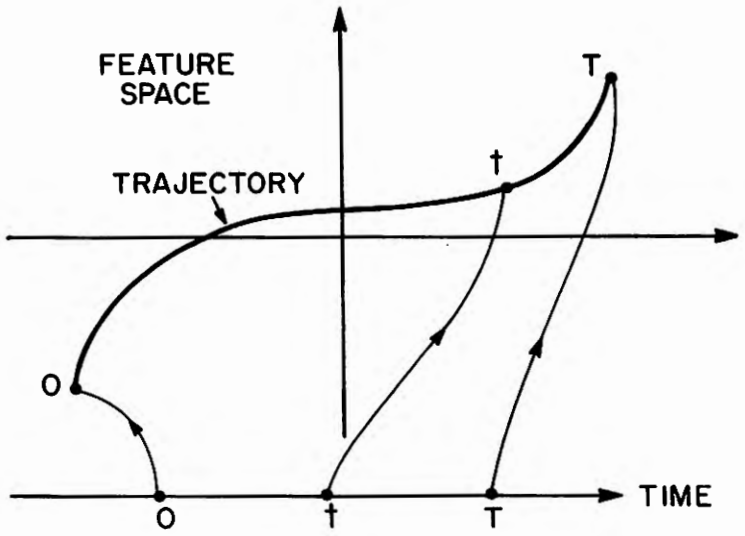


Figure 1a. Trajectory.

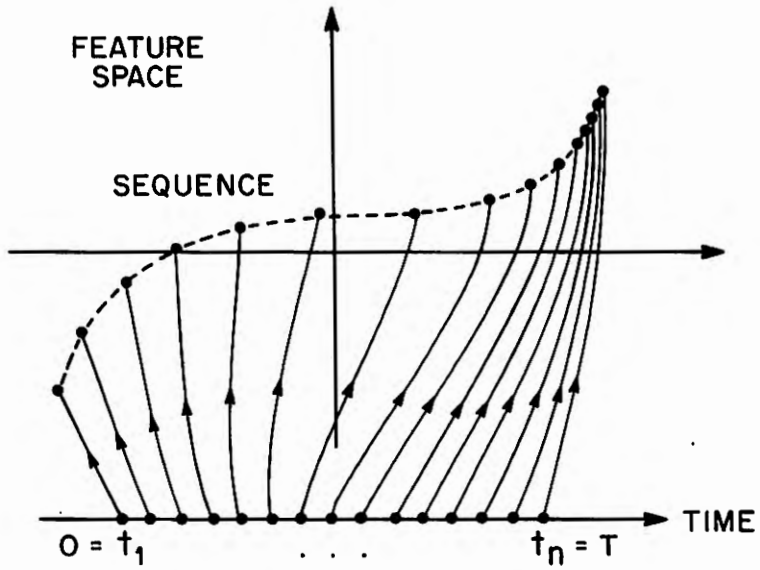


Figure 1b. Sequence derived from trajectory.

strings of computer science, in which the corresponding source of variation is deletion and insertion of units. This similarity is based on the following correspondence:

Compression–Expansion		Deletion–Insertion
Compress 2 units into 1	↔	Delete 1 unit
Expand 1 unit into 2	↔	Insert 1 unit

More generally,

Compress $(k+1)$ adjacent units into 1	↔	Delete k adjacent units
Expand 1 unit into $(k+1)$ adjacent units	↔	Insert k adjacent units

It is, however, frequently overlooked that the difference in meaning between compression–expansion and deletion–insertion leads to significantly different definitions of distance between sequences. We shall make the difference very clear below, and illustrate how to use both types of change at the same time when comparing sequences.

In fields where time-warping is used, the basic objects of interest are generally continuous trajectories, so it is natural in concept, though impossible in practice, to compare the trajectories directly. While the conversion of trajectories to sequences by sampling circumvents the practical difficulty, many of the ideas of time-warping can be expressed most naturally in a continuous setting. In this chapter we first develop continuous time-warping, and then systematically "discretize" it, i.e., formulate discrete analogues to all concepts and definitions involved. This appears to be the first paper in which continuous time-warping is formulated in a fully symmetric manner, and the first in which the discretization process is systematically examined and a variety of alternative discretizations specified. This approach provides a full justification for some edge weights (such as "$\frac{1}{2}$, 1, $\frac{1}{2}$"), which have been widely used without a fully satisfying rationale.

A method of sequence comparison is symmetric, in the sense used above, if comparing **a** with **b** gives the "same" result as comparing **b** with **a**, that is, the distances are the same and the time-warping of **b** onto **a** is the inverse of the time-warping of **a** onto **b**. Although the methods of sequence comparison in speech recognition are often deliberately asymmetric, treating the "stored template" utterance differently from the utterance to be recognized, our development is almost entirely limited to methods that are symmetric. There are several reasons for this.

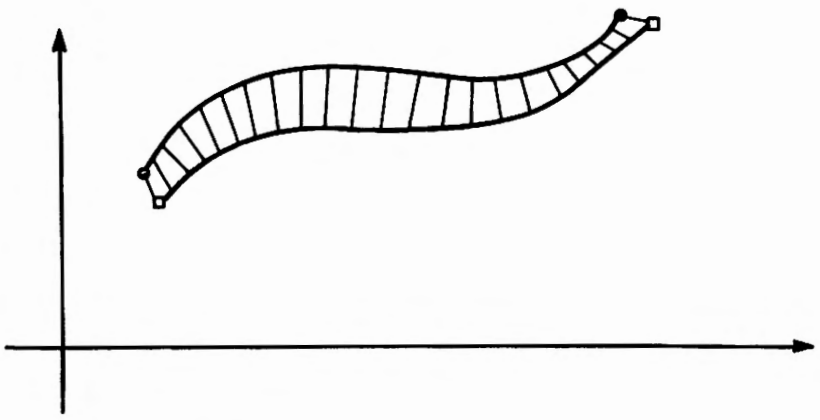


Figure 2a. Intuitive idea of continuous time-warping.

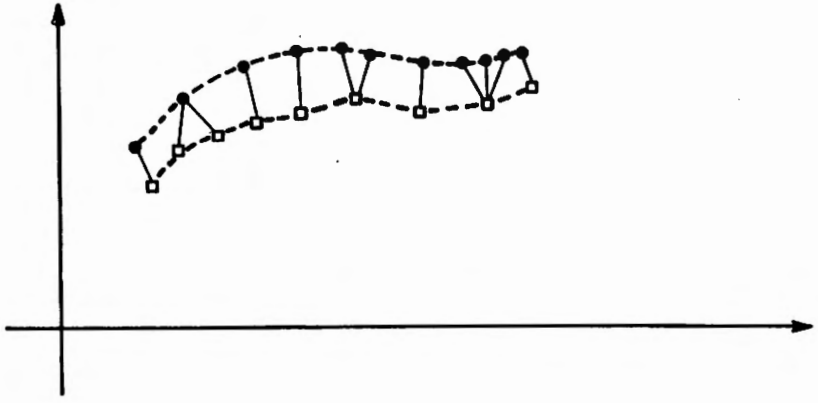


Figure 2b. Intuitive idea of discrete time-warping.

1. One purpose of this paper is to clarify the central difference between the comparison methods of molecular biology and those of speech processing, i.e., the differences between what we now distinguish as deletion–insertion and compression–expansion. The question of symmetry or asymmetry is not important for this purpose, and the symmetric approach is more convenient to work with, and familiar in

2. Another purpose of this paper is a proposed speech application for which symmetric comparison is desired, specifically, the comparison of related utterances so as to study timing variability of normal speech. The data may consist either of x-ray microbeam recordings of articulator motion (as described, e.g., in Fujimura (1981)), or conventional sound wave analyses as used in speech processing.
3. The chief reason for asymmetric comparison in speech recognition lies in the the mild improvement obtained by distinguishing between the stored template and unidentified current utterance. Even in speech recognition, however, there are other uses for comparison in which the desirability of asymmetry is not so clear, e.g., combining of utterances to form an "average" template. Thus, insight into symmetric methods may perhaps be of value even for speech recognition.

In Secs. 2 and 3 we formalize the notion of a time-warping as a "linking" that connects the time scales of the two trajectories or sequences. In the discrete case, the linking concept is similar to the "trace" concept used with deletion–insertion comparisons (see, for example, Chapter 1). In fact, a discrete linking is precisely analogous to a trace, and the differences between linking and trace reflect the differences between compression–expansion and deletion–insertion. In Secs. 4 and 5 we define the length of a linking, and then define distance between two trajectories as the minimum possible length of any linking between them. There is quite a variety of different ways to discretize the concept of length, which lead to mildly different discrete concepts. We explore many of these, including some that have not previously been discussed.

In Sec. 6 we explain the most important difference between compression–expansion and deletion–insertion, namely, the difference between the length of a linking and the length of a trace. Linking length does not use deletion–insertion costs as trace length does, only substitution costs. On the other hand, linking length uses another distinctive element called time-weights, which multiply the substitution costs. In Sec. 7 we explain how compression–expansion and deletion–insertion can be combined into a single potentially useful method, by incorporating both deletion–insertion costs and time-weights in the same comparison.

In Sec. 8, we note that a time-warping between two trajectories may be seriously misleading when the interval at which the trajectories are sampled is large in comparison to the differences between them, and we introduce a new method called *interpolation time-warping* to remedy this difficulty. In Secs. 9 and 10, stimulated by the asymmetric definition of Rabiner and Wilpon (1979, 1980), we give a symmetric definition of a weighted average between two trajectories or two sequences. Averaging is useful in forming a single "typical" sequence that is intended to represent a set of several similar sequences.

We note that when time-warping is applied, numerous related problems need to be dealt with that may not be part of the time-warping itself. These

problems include choice of distance function (called w below) in the feature space, local constraints on the time-warping function, and finding where the trajectories begin and end (finding where speech utterances begin and end is surprisingly difficult). The solutions to these problems depend strongly on the domain of application. This paper is devoted to the central time-warping concept itself, and does not deal with problems such as those mentioned.

For information about methodology and the use of time-warping in recognition of isolated words, the reader may consult papers such as Itakura (1975), White (1978), Sakoe and Chiba (1978), Myers, Rabiner, and Rosenberg (1980), Rabiner, Rosenberg and Levinson (1978), and White and Neely (1976). For methodology and the use of time-warping in recognition of connected speech, see Chapter 5 and papers such as Bridle and Brown (1979), Rabiner and Schmidt (1980), and Myers and Rabiner (1981a, 1981b). In addition, a volume of reprints, Dixon and Martin (1979), contains many valuable papers in this field. For applications of time-warping to gas chromatography, see Reiner *et al.* (1979, 1978, 1969). For applications to handwriting recognition and related topics, see Fujimoto *et al.* (1976), Burr (1979, 1980, 1981), and Yasuhara and Oka (1977).

2. TIME-WARPING IN THE CONTINUOUS CASE

In speech processing, gas chromatography, bird song, and other potential applications of sequence comparison, the underlying objects of interest are basically continuous functions $\mathbf{a}(t)$, $\mathbf{b}(t)$, etc., of a continuous variable t, which is often time. Also, the values of the functions lie in a several-dimensional space which we shall call the *feature space.* Thus each object of interest is a continuous *trajectory* or curve through feature space, as shown in Fig. 1(a), in which each point on the curve corresponds to a particular value of the variable t. For practical manipulation, these trajectories are ordinarily converted into sequences by sampling the values of t, as shown in Fig. 1(b). Geometrically, this corresponds to describing the trajectory by a series of points on it.

By way of example, we mention that in speech processing, the dimensionality of the feature space is often in the range from 6 to 15. The ith coordinate of $\mathbf{a}(t)$ might indicate the power present in a speech utterance in the ith frequency band at time t (using a short-time spectral analysis). Alternatively, it might indicate the ith linear predictor coefficient at time t.

Conceptually, time-warping applies most directly to comparisons of continuous trajectories. It has seldom been discussed in this domain, however, because for practical computation it is always used with sequences. We start, however, by discussing time-warping and its uses in the continuous case, for the conceptual guidance this discussion provides in the discrete case.

Two trajectories

are said to be connected by an [approximate] continuous time-warping if they traverse [approximately] the same curve in feature space in the same direction, though at possibly very different rates; for example, $\mathbf{a}(u)$ may proceed slowly along an early portion of the curve and quickly along a later portion, while $\mathbf{b}(v)$ might do the reverse.

Geometrically, the idea of a time-warping is that each point in one trajectory corresponds to some specific point in the other. One way to visualize this is illustrated in Fig. 2(a), in which corresponding points are connected by line segments. If $\mathbf{a}(u)$ corresponds to $\mathbf{b}(v)$, we say u is *linked* to v. The correspondence between the trajectories is the central idea of time-warping.

More formally, we say (see Fig. 3(a)) that $\mathbf{a}(u)$ and $\mathbf{b}(v)$ are *connected by an [approximate] continuous time-warping* $(\mathbf{u}_0, \mathbf{v}_0)$ if $\mathbf{u}_0(t)$ and $\mathbf{v}_0(t)$ are strictly increasing functions defined for $0 \leqq t \leqq T$ such that

$$\mathbf{a}(\mathbf{u}_0(t)) = \mathbf{b}(\mathbf{v}_0(t)) \quad [\text{or } \mathbf{a}(\mathbf{u}_0(t)) \cong \mathbf{b}(\mathbf{v}_0(t))].$$

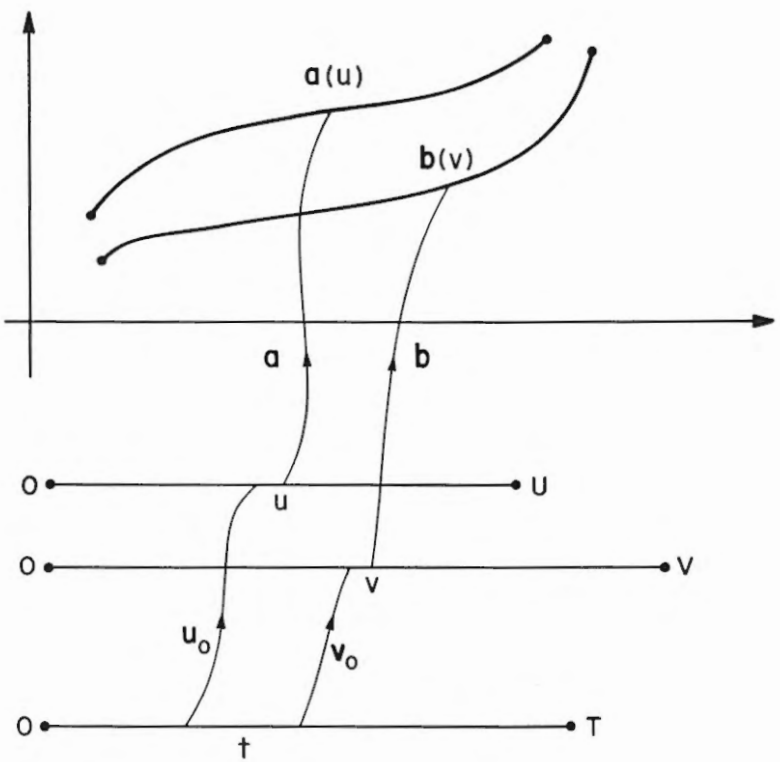


Figure 3a. Continuous time-warping.

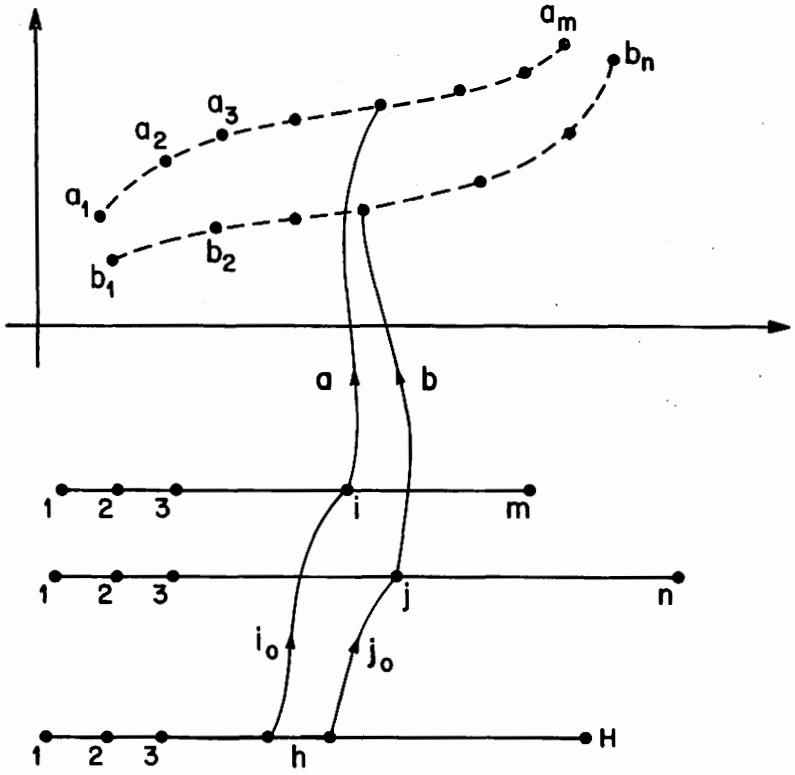


Figure 3b. Discrete time-warping.

Some constraint is generally needed to ensure that the time-warping does not degenerate to some tiny part of the curves involved. For example, the constraint might be that

$$\mathbf{u}_0(0) = 0, \qquad \mathbf{v}_0(0) = 0,$$
$$\mathbf{u}_0(T) = U, \qquad \mathbf{v}_0(T) = V,$$

though a weaker constraint could also be used. The word "approximate" is frequently omitted even when the approximate sense is intended, and the word "continuous" is generally omitted since it is obvious from context.

In this formulation the time-warping correspondence between the two trajectories is mediated by linking the two time-scales u and v. If $u = \mathbf{u}_0(t)$ and $v = \mathbf{v}_0(t)$, we shall say that *u and v are linked by* $(\mathbf{u}_0, \mathbf{v}_0)$ *at* t. Thus, points in the

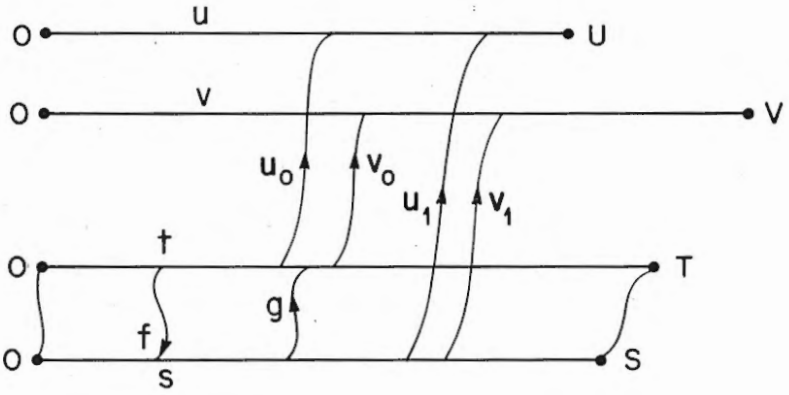


Figure 4. Arbitrariness of time scale.

It is necessary to recognize that the scale on which the parameter t occurs is arbitrary, has no intrinsic meaning, and can be freely distorted. In particular (see Fig. 4), suppose that f is any strictly increasing continuous function for which $f(0) = 0$, and let $s = f(t)$, $S = f(T)$. Let g be the inverse function of f (that is $g(f(t)) = t$, $f(g(s)) = s$), so that $t = g(s)$, $T = g(S)$. Define another time-warping $(\mathbf{u}_1, \mathbf{v}_1)$ by

$$\mathbf{u}_1(s) = \mathbf{u}_0(g(s)), \qquad \mathbf{v}_1(s) = \mathbf{v}_0(g(s)).$$

All we have done is distort the arbitrary scale for t into another arbitrary scale for s. The time-warping $(\mathbf{u}_1, \mathbf{v}_1)$ gives the same correspondence between the two trajectories as $(\mathbf{u}_0, \mathbf{v}_0)$, since if u and v are linked by $(\mathbf{u}_0, \mathbf{v}_0)$ at t, then it is easy to verify that they are linked by $(\mathbf{u}_1, \mathbf{v}_1)$ at $s = f(t)$.

We shall call two time-warpings *equivalent* if they induce the same linking (throughout the entire trajectories). We note without proof that any two equivalent time-warpings must be related in the manner just described. We think of equivalent time-warpings as essentially the same, and differing only in external form, not in any substantive way. This view will have important implications below.

In its various applications, time-warping is used as a method to help overcome the variability among nominally identical trajectories. Conceptually, we can think of a trajectory as composed of two aspects: One is the curve, by which we mean the points swept out; the other is the time pattern, by which we mean the rate at which the curve is followed. The time-warping we construct between two trajectories displays the difference between them in terms of these aspects: $\mathbf{u}_0$ and $\mathbf{v}_0$ compare the time patterns, while the distance is a summary measure of how much the curves differ.

Note that the methods that are used to calculate a time-warping between two trajectories must frequently be applied when the trajectories are, in fact, entirely different and unrelated, e.g., when comparing an observed trajectory with many stored trajectories in order to identify it. Thus, although the basic concept assumes the existence of an approximate time-warping, the methods for calculation must not rely too heavily on this assumption.

Speech processing has generally rested, of course, on the basic assumption that two trajectories of the same word or phrase are connected by an approximate time-warping. While this assumption is reasonable and has been the basis for a great deal of fruitful work, systematic violations are known to occur. For instance, more emphatic pronunciation generally produces not only an increase in duration, but also an "amplification" of the vocal gestures involved. This effect can be seen most clearly in articulatory data, as expansion of some portion of the curve, but formant trajectories also show it plainly. In the filter-bank or linear-prediction feature spaces, such phenomena are equally present, though harder to visualize. Obviously, in such a case the usual time-warping comparison will produce a distance measure that is "too large," because it does not allow for trajectory differences that leave the word or phrase unchanged. (Also, it is observed empirically that when "amplification" of a curve occurs, the usual procedures yield a time-warping that differs quite strongly from our intuitive notion of what it should be.) Such problems are doubtless among the reasons that speech recognition has been such a challenging problem.

3. TIME-WARPING IN THE DISCRETE CASE

To work with the continuous trajectories in practice, one standard approach is to convert them to sequences of points in feature space by sampling (see Fig. 1(b)). To convert $\mathbf{a}(u)$, it is sampled at some suitable set of discrete values $u_1, \ldots, u_m$, and $\mathbf{a}(u)$ is represented by the sequence $(\mathbf{a}(u_1), \ldots, \mathbf{a}(u_m))$. We shall use $a_i = \mathbf{a}(u_i)$ and $\mathbf{a} = a_1 \ldots a_m$. In a similar manner, $\mathbf{b}(v)$ is represented by its values at $v_1, \ldots, v_n$, namely by $\mathbf{b} = b_1 \ldots b_n = (\mathbf{b}(v_1), \ldots, \mathbf{b}(v_n))$.

It is also necessary to convert the time-warping concept from trajectories to sequences. This could be done in more than one way, but we follow the usual definition, which seems very plausible. Following the definition, we justify certain parts of it. As illustrated in Fig. 2(b) and 3(b), two sequences

$$\mathbf{a} = a_1 \ldots a_m \qquad \text{and} \qquad \mathbf{b} = b_1 \ldots b_n$$

with entries in the feature space are said to be *connected by an [approximate] discrete time-warping* $(\mathbf{i}_0, \mathbf{j}_0)$ if $\mathbf{i}_0(h)$ and $\mathbf{j}_0(h)$ are weakly increasing integer functions defined for $1 \leqq h \leqq H$ satisfying a "continuity constraint" (see below) such that

for all h. Each value of h corresponds to a line in Fig. 2(b) that connects a point in one sequence to a point in the other. To avoid the possibility that the time-warping degenerates to a tiny part of the sequences, we can use the constraint

$$\mathbf{i}_0(1) = 1, \qquad \mathbf{j}_0(1) = 1,$$
$$\mathbf{i}_0(H) = m, \qquad \mathbf{j}_0(H) = n,$$

though a weaker constraint could also be used. The word "approximate" is frequently omitted, even where the approximate sense is intended, and the word "discrete" is generally omitted since it is obvious from context.

The time-warping correspondence between the two sequences is mediated by linking what are in effect discrete time scales, i and j. If $i = \mathbf{i}_0(h)$ and $j = \mathbf{j}_0(h)$, we shall say that i and j are linked by $(\mathbf{i}_0, \mathbf{j}_0)$ at h. The points in the sequence correspond in the time-warping if their times are linked.

Figure 5 illustrates another representation of a discrete time-warping that is particularly important in connection with practical computation. For a given

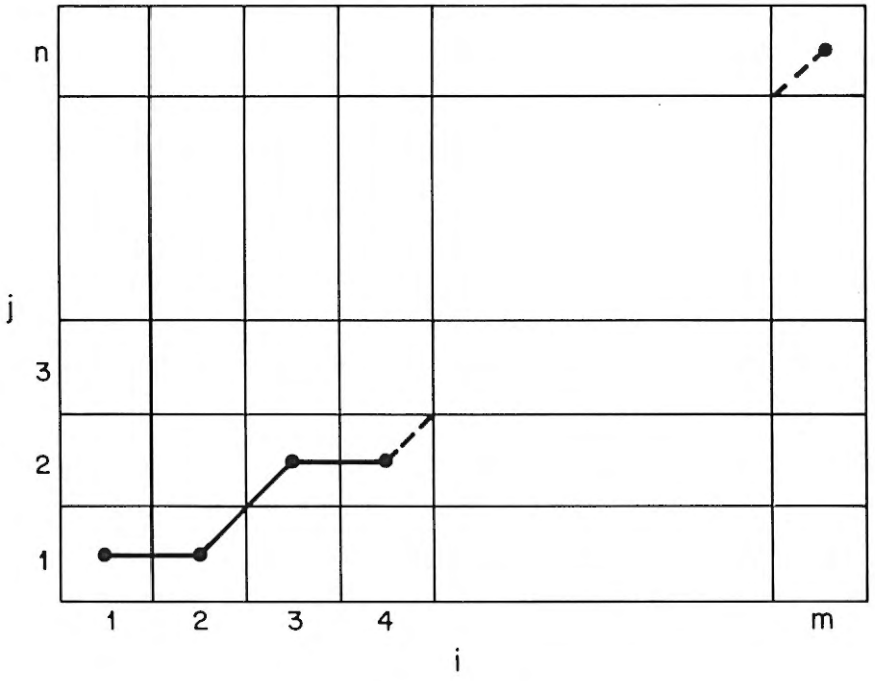


Figure 5. Computational array.

time-warping $(\mathbf{i}_0, \mathbf{j}_0)$, each h corresponds to one cell in the computational array, namely, to cell $(\mathbf{i}_0(h), \mathbf{j}_0(h))$. If each such cell is indicated by a dot, and adjacent cells are connected by lines, the entire warping can be visualized as a path through the array.

We describe two different continuity constraints and justify them below. Each description is in terms of the vector or step between adjacent points of the time-warping in Fig. 5, that is, in terms of $(\Delta\mathbf{i}_0, \Delta\mathbf{j}_0)$, where Δ is defined by $\Delta\mathbf{i}_0(h) = \mathbf{i}_0(h) - \mathbf{i}_0(h-1)$. The first and most commonly used continuity constraint (see Fig. 6(a)) is

$$(\Delta\mathbf{i}_0, \Delta\mathbf{j}_0) = \begin{cases} (1, 0) & \text{or} \\ (1, 1) & \text{or} \\ (0, 1). \end{cases}$$

We remark that the step (1, 0) indicates a time-compression from **a** to **b** that reduces the number of units by one: If there are k adjacent steps of this type, they constitute compression of $k + 1$ units into one. (Readers accustomed to deletion–insertion comparison are reminded that this step does *not* correspond to a deletion. The difference between compression and deletion will be discussed later. In the present notation, a single deletion could be indicated by a step of (2, 1).) Similarly, the vector (0, 1) indicates a time expansion from **a** to **b** that increases the number of units by one.

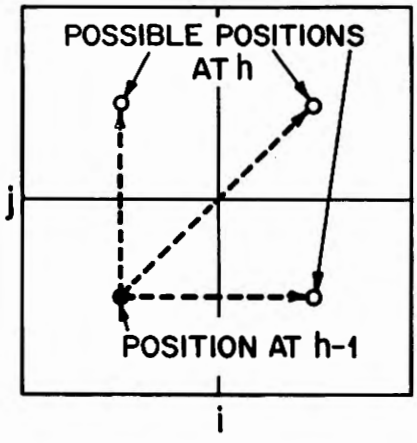


Figure 6a. First continuity constraint.

A second continuity constraint (see Fig. 6(b)), which is used in Chapter 5 by Hunt, Lennig, and Mermelstein and is due to Mermelstein, is

$$(\Delta i_0, \Delta j_0) = \begin{cases} (2, 0) & \text{or} \\ (1, 1) & \text{or} \\ (0, 2). \end{cases}$$

Under this constraint, only cells (i, j) for which $i + j$ is even are used, since the other pairs are skipped over. Also, it is not hard to see that every time-warping uses exactly the same number of pairs (i, j) (that is, same value of H), in contrast to the first constraint. Still other continuity constraints have been used also, but we do not consider them here.

Sometimes weights (most often $\frac{1}{2}$, 1, $\frac{1}{2}$) are associated with the alternative steps of the first constraint, for use in evaluating the length of a time-warping When we discuss lengths of time-warping later on, weights will arise naturally of their own accord; this fact and the values of these weights are a topic of interest to us.

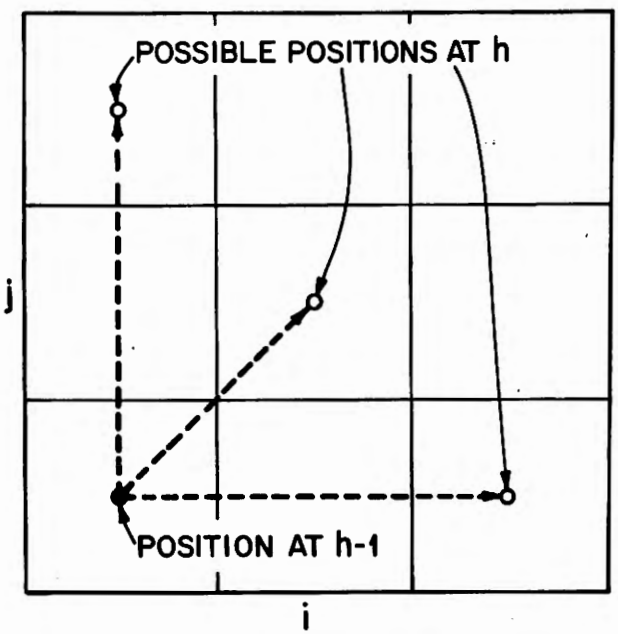
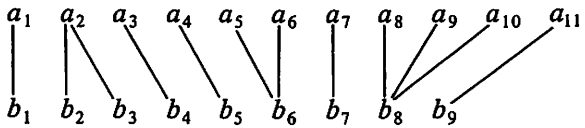


Figure 6b. Second continuity constraint.

Now, however, we simply wish to explain the continuity constraints. In the continuous case, $\mathbf{u}_0$ and $\mathbf{v}_0$ are constrained by monotonicity constraints, and also by continuity conditions that insured that no part of either trajectory can be skipped, i.e., there is no insertion or deletion. The first continuity constraint is exactly what we need to insure monotonicity of $\mathbf{i}_0$ and $\mathbf{j}_0$ and to avoid insertion and deletion. If we are willing to restrict the time-warping to points (i, j) for which $i+j$ is even (i.e., squares of only one color on a checkerboard), then the second continuity constraint is obtained in a similar manner. While the restriction to even values of $i+j$ appears to discard some fine-grain information, it reduces computation time by a factor of two, and its use is favored by Hunt, Lennig, and Mermelstein, partly because of the property that H is the same for all time-warpings. If the sampling rate for converting trajectories into sequences is increased by $\sqrt{2}$, this would appear to balance out the loss of information effectively while restoring the computation time, so the choice of continuity constraint should depend on subtler considerations.

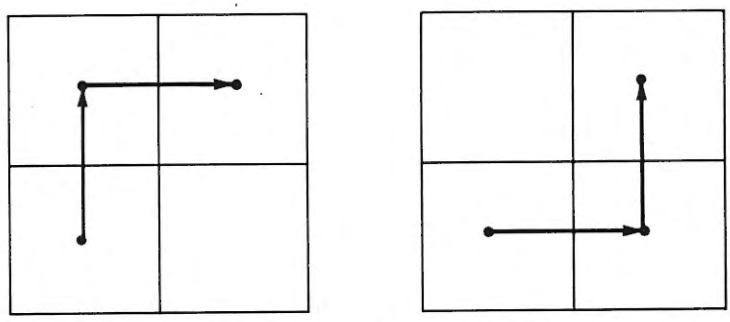
To display a discrete time-warping pictorially, we can use a diagram like the one shown in Fig. 2(b), where each line corresponds to one value of h. If a_i is connected to h consecutive terms $b_j, \ldots, b_{j+h-1}$, this indicates that a region of $\mathbf{a}(u)$ around a_i corresponds in this time-warping to a region of $\mathbf{b}(v)$ around $b_j, \ldots, b_{j+h-1}$. If $u_1, u_2, \ldots$ and $v_1, v_2, \ldots$ are points in time and are regularly spaced using the same interval for the u_i and the v_j, this correspondence indicates that the changes in $\mathbf{a}(u)$ around time u_i occur rapidly and time must be stretched to match the corresponding changes in $\mathbf{b}(v)$ over the interval from v_j to v_{j+h-1}, which occur slowly. Of course, if the multiple connections go the other way, then a similar interpretation holds in reverse.

A diagram somewhat like Fig. 2(b) can be presented more simply:

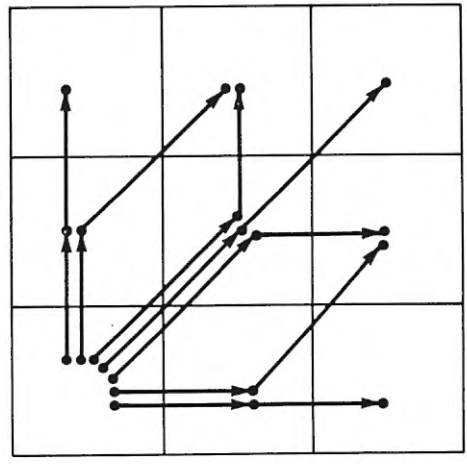


Discrete time-warping diagrams like this are very similar to trace diagrams (see, e.g., Chapter 1). However, such diagrams for symmetric time-warping differ from trace diagrams in two ways. (These remarks do not fully apply to diagrams for asymmetric time-warping, which is frequently used in speech processing.)

1. In a symmetric time-warping diagram, one term of a sequence may be connected to several terms of the other sequence, while in a trace diagram each term can be connected to at most one other term.
2. In a symmetric time-warping diagram, every term is connected to at least one other term, while in a trace diagram not every term need have a connection (i.e., terms that are insertions or deletions).



FORBIDDEN TWO-STEP PATTERNS



PERMITTED TWO-STEP PATTERNS

Figure 7. An additional constraint.

Additional constraints on the time-warping are common in speech research. Characteristically, they refer to the values of $\mathbf{i}_0$ and $\mathbf{j}_0$ at three or more consecutive values of h. The simplest, least restrictive constraint of this type simply forbids an "N"-shaped configuration in the time-warping diagram like those shown here:

Forbidden $\left\{ \begin{matrix} a & a \\ \diagdown & \diagup \\ b & b \end{matrix} \right.$, $\begin{matrix} a & a \\ \diagup & \diagdown \\ b & b \end{matrix}$;

that is, if a term has multiple lines, the terms at the other ends of these lines must not have multiple connections. Other ways of describing this constraint are shown in Fig. 7.

4. DISTANCE AND LENGTH IN THE CONTINUOUS CASE

The basic idea of time-warping is that replications of nominally the same trajectory will trace out approximately the same curve, but with varying time patterns. To measure the extent to which two trajectories $\mathbf{a}(u)$ and $\mathbf{b}(v)$ deviate from having this ideal relationship, we will first define the *length* $\bar{d}(\mathbf{u}_0, \mathbf{v}_0)$ of any given time-warping $(\mathbf{u}_0, \mathbf{v}_0)$ in a suitable way, as the distance between corresponding points in the two trajectories pooled somehow over the entire trajectories. Of course $\bar{d}$ depends on $\mathbf{a}(u)$ and $\mathbf{b}(v)$ as well as on $\mathbf{u}_0$ and $\mathbf{v}_0$, but we omit this dependence from the notation, for simplicity. Once length is defined, then the distance is given by

$$d(\mathbf{a}(u), \mathbf{b}(v)) = \min_{\text{all } (\mathbf{u}_0, \mathbf{v}_0)} \bar{d}(\mathbf{u}_0, \mathbf{v}_0);$$

that is, the distance is the length of the shortest possible time-warping.

To define the length of $\bar{d}(\mathbf{u}_0, \mathbf{v}_0)$, we first need a way to measure how far apart corresponding points $\mathbf{a}(\mathbf{u}_0(t))$ and $\mathbf{b}(\mathbf{v}_0(t))$ are for a fixed t. We shall assume that a distance function $w[a, b]$ suitable for this purpose has been defined on the feature space. This function could be simple Euclidean distance, or weighted Euclidean distance, or a more complicated function in which the distance is sensitive to position in the feature space.

To pool this distance over the whole trajectory, the obvious definition for $\bar{d}(\mathbf{u}_0, \mathbf{v}_0)$ might seem to be

$$\int_0^T w[\mathbf{a}(\mathbf{u}_0(t)), \mathbf{b}(\mathbf{v}_0(t))] \, dt.$$

This definition, however, is incorrect. Recall that we defined time-warpings to be equivalent if they induce the same linking between the trajectories, and that we consider equivalent time-warpings as essentially the same. We want a definition for which equivalent time-warpings have the same length. Using the definition above, however, equivalent time-warpings can result in quite different lengths. To see this, suppose we generate another time-warping equivalent to $(\mathbf{u}_0, \mathbf{v}_0)$, as in Fig. 4, by using $s = f(t)$ with f an increasing continuous function such that $f(0) = 0$ and $f(S) = T$. Writing the integral parallel to the one above but for $(\mathbf{u}_1, \mathbf{v}_1)$ instead, and then transforming it by $s = f(t)$, we have

$$\int_0^S w[\mathbf{a}(\mathbf{u}_1(s)), \mathbf{b}(\mathbf{v}_1(s))] \, ds = \int_0^T w[\mathbf{a}(\mathbf{u}_0(t)), \mathbf{b}(\mathbf{v}_0(t))] f'(t) dt.$$

This is exactly the same as the integral for $(\mathbf{u}_0, \mathbf{v}_0)$ *except for the factor* $f'(t)$, which can be virtually *any positive function.* Obviously, different choices of f' yield different values for the integral, so this definition is not invariant.

The arbitrariness of the integral has a precise geometrical interpretation: The integral runs along the two trajectories at an arbitrary rate that is determined by the arbitrary time scale on which t is measured. If we rush along the trajectories (this will be the case for $(\mathbf{u}_1, \mathbf{v}_1)$ if $g'(s)$ is large, $f'(t)$ small, and S small), then the integral will be small. If we go along the trajectories slowly (corresponding to the reverse situation), the integral will be large. Furthermore, even if the overall rate is the same for two time-warpings, we can still rush along the curves where they are far apart and go slowly where they are close together, in order to get a small value, or use the reverse strategy to get a large value.

Once the problem is stated, there is an obvious solution. The trajectories themselves have natural meaningful time scales, and we should use these time scales to weight each infinitesimal portion of the integral by a weight that corresponds to how long the trajectories linger there. Specifically, suppose u and v are linked by $(\mathbf{u}_0, \mathbf{v}_0)$ at t. The trajectory $\mathbf{a}(u)$ spends time $du = \mathbf{u}_0'(t)\, dt$ in the infinitesimal region around u, and the trajectory $\mathbf{b}(v)$ spends time $dv = \mathbf{v}_0'(t)\, dt$ around v. If we were willing to accept an asymmetric formulation, we could use either $\mathbf{u}_0'(t)$ or $\mathbf{v}_0'(t)$ as the weighting function. Let us disregard the asymmetry for a moment, and consider the use of $\mathbf{u}_0'(t)$. It gives the integral

$$\int_0^T w[\mathbf{a}(\mathbf{u}_0(t)),\ \mathbf{b}(\mathbf{v}_0(t)]\mathbf{u}_0'(t) \quad dt.$$

To test for invariance, we generate $(\mathbf{u}_1, \mathbf{v}_1)$ in the same way as before, and consider its corresponding integral,

$$\int_0^S w[\mathbf{a}(\mathbf{u}_1(s)),\ \mathbf{b}(\mathbf{v}_1(s))]\ \mathbf{u}_1'(s)\ ds.$$

Consider the new factor $\mathbf{u}_1'(s)$. We have

$$\mathbf{u}_1'(s) = \frac{d}{ds}\,\mathbf{u}_0(g(s)) = \mathbf{u}_0'(g(s))g'(s).$$

Now differentiating $f(g(s)) = s$,

$$f'(g(s)) \cdot g'(s) = 1, \qquad g'(s) = \frac{1}{f'(t)}\ .$$

Therefore if we transform by $s = f(t)$, the preceding integral equals

$$\int_0^T w[\mathbf{a}(\mathbf{u}_0(t)),\ \mathbf{b}(\mathbf{v}_0(t))] \cdot \frac{\mathbf{u}_0'(t)}{f'(t)} \cdot f'(t)\, dt = \int_0^T w[\cdots]\, \mathbf{u}_0'(t)\, dt.$$

This is the same as the integral corresponding to $(\mathbf{u}_0, \mathbf{v}_0)$, so the new definition of length has the desired invariance property.

Use of $\mathbf{u}_0'(t)$ as the weighting function effectively means that we run along the $\mathbf{a}(u)$ trajectory at the rate set by its time scale, and along the $\mathbf{b}(v)$ trajectory however the correspondence determines. Use of $\mathbf{v}_0'(t)$ reverses the roles of the two trajectories. Either of these gives to the definition of length the invariance property, but neither one treats the two trajectories symmetrically. (This asymmetric approach, incidentally, is used in much speech-recognition work, where the time scale of the unknown utterance trajectory is used to form the distance.)

To give a symmetric formulation that is invariant, we must use a weighting function that combines $\mathbf{u}_0'(t)$ and $\mathbf{v}_0'(t)$ in a symmetric way. The most obvious possibility is the average, $(\mathbf{u}_0'(t) + \mathbf{v}_0'(t))/2$ (or alternatively, the sum). This means that we run along the trajectories at the average of the rates set by their two time scales. Other possibilities that provide both invariance and symmetry include the geometric mean, $(\mathbf{u}_0'(t)\mathbf{v}_0'(t))^{1/2}$, the rth power mean for any r, that is,

$$\left[\frac{\mathbf{u}_0'(t)^r + \mathbf{v}_0'(t)^r}{2}\right]^{1/r},$$

and still more general types of mean value. The case $r = 2$ can be given an arc-length interpretation, and turns out, after manipulation, to be the same as a formula from Myers (1980), which is discussed below.

Generalizing in another direction, a weighted combination of $\mathbf{u}_0'(t)$ and $\mathbf{v}_0'(t)$ with weights U and V (recall that $U = \mathbf{u}_0(T)$, $V = \mathbf{v}_0(T)$), or $1/U$ and $1/V$, or $f(U)$ and $f(V)$ for any function f, is also symmetric and invariant. Using weights $1/U$ and $1/V$ leads to an attractive weighting function $(\mathbf{u}_0'(t)/U + (\mathbf{v}_0'(t)/V)$. Of course, weights could also be incorporated into the generalized means as well.

Lacking a convincing argument for any particular one of these formulations, we choose the ordinary average merely for simplicity. Thus for the remainder of this paper, $d(\mathbf{u}_0, \mathbf{v}_0)$ is formed by minimizing the following length over $(\mathbf{u}_0, \mathbf{v}_0)$:

$$\bar{d}(\mathbf{u}_0, \mathbf{v}_0) \equiv_{\text{def}} \int_0^T w[\mathbf{a}(\mathbf{u}_0(t)),\ \mathbf{b}(\mathbf{v}_0(t)]\ \frac{\mathbf{u}_0'(t) + \mathbf{v}_0'(t)}{2}\, dt.$$

4.1 Comparison with Other Continuous Formulations

In the many papers that apply time-warping to speech processing, there have been very few discussions of the continuous time-warping problem. In published papers, we note a brief discussion in Velichko and Zagoruyko (1970), a brief mention in Sakoe and Chiba (1971), and a discussion limited largely to the one-dimensional feature space in Levinson (1981). In addition, we note a more extensive discussion in an unpublished paper by Myers (1980).

Sakoe and Chiba (1971) present the following integral (our notation),

$$\int_0^U w[\mathbf{a}(u), \mathbf{b}(\mathbf{f}(u))] \quad du.$$

where u is the time parameter for utterance $\mathbf{a}$, and $\mathbf{f}(u)$ (which describes the time-warping) corresponds to $\mathbf{v}_0(\mathbf{u}_0^{-1}(u))$, in our notation. Their integral is essentially the same as our first correct (but asymmetric) integral given above, since the two integrals are connected by an elementary change of variables, $u = \mathbf{u}_0(t)$. They propose minimizing this integral by choice of $\mathbf{f}$. Although their integral is not symmetric in the two utterances, they then state that minimization problems of this type can be "very effectively solved by dynamic-programming technique as follows," and proceed to present a symmetric version of the discrete time-warping problem, but do not indicate how the discrete formulation is derived from the continuous one.

The discussion by Velichko and Zagoruyko (1970) is harder to summarize, because it is less precisely stated. After developing a discrete version of time-warping, they state that "in the continuous approximation, the sum is substituted by the integral"

$$\int_{(\ell)} b(\ell) d\ell,$$

where the integral is taken along a curve in the (u, v)-plane (our notation), ℓ appears to be arc length along the curve, and $b(\ell)$ appears to be a measure of similarity between $\mathbf{a}(u)$ and $\mathbf{b}(v)$ at point ℓ on the curve. Presumably, $b(\ell)$ is intended to be analogous to their discrete measure of similarity ρ^2 defined shortly before. The curve, of course, describes the time-warping, which they refer to as a time normalization. They then argue for introduction (into the integral) of a weighting factor $f(\gamma)$ such as $f(\gamma) = \cos(2\gamma)$, where γ is the angle between the 45° line and the tangent to the curve at point ℓ, thus yielding

$$\int_{(\ell)} f(\gamma) b(\ell)\, d\ell.$$

They propose finding the curve that maximizes this integral, subject only to the constraint that the curve denote a monotonic increasing function, and describe this maximization as a variational problem. After this they "reformulate (their) problem for the discrete case," but give no details connecting the continuous and discrete formulations.

The discussion by Levinson (1981) is largely subsumed by that within Appendix I of Myers (1980), which we now discuss. Myers presents the following integral (notation partly changed to ours):

$$\int_0^U w[\mathbf{a}(u), \mathbf{b}(\mathbf{f}(u))]\,\tilde{W}(u, \mathbf{f}(u), \dot{\mathbf{f}}(u))\,du.$$

Note that this is the same as Sakoe and Chiba's integral, except for the introduction of the weighting function $\tilde{W}$. The use of a weighting factor of this form appears to be largely based on a fact introduced by Myers, namely, that this integral fits within the framework of a much-studied problem in the calculus of variations. Myers introduces the solution from that field, which is a differential equation for the time-warping curve $v = \mathbf{f}(u)$ in the (u, v) plane, and then proceeds to discuss choice of $\tilde{W}$. He drops the dependence of $\tilde{W}$ on u and $\mathbf{f}(u)$ "since all points in the $[(u, v)]$ plane should be weighted equally," and proposes as one logical choice for $\tilde{W}$ the form

$$\tilde{W}(\dot{\mathbf{f}}(u)) = \sqrt{1 + \dot{\mathbf{f}}(u)^2},$$

since $\tilde{W}(\dot{\mathbf{f}}(u))\,du$ then becomes the differential of arc length along the time-warping curve. Thus he obtains

$$\int_0^U w[\mathbf{a}(u), \mathbf{b}(\mathbf{f}(u))]\sqrt{1 + \dot{\mathbf{f}}(u)^2}\,du.$$

He points out that this can be thought of as the line integral of w with respect to arc length over the time-warping curve (and thus obtains an integral very similar to that of Velichko and Zagoruyko, though he does not make the connection or cite their paper). He attempted to find a numerical solution to the differential equation for his choice of $\tilde{W}$, but indicates that this turned out to be difficult. He does not make any detailed connection between the continuous and discrete versions of the time-warping problem.

Myer's integral turns out to be symmetric in the two utterances **a** and **b**, as we can see from the arc-length formulation, although his definition is not phrased in a symmetric manner and he does not consider the matter of symmetry. His integral above can easily be transformed, using the elementary change of variables $u = \mathbf{u}_0(t)$, into the symmetric form

$$\int_0^T w[\mathbf{a}(\mathbf{u}_0(t)), \mathbf{b}(\mathbf{v}_0(t))] \left[\frac{\mathbf{u}_0'(t)^2 + \mathbf{v}_0'(t)^2}{2}\right]^{1/2} dt,$$

which was one of the symmetric forms we described above.

5. DISTANCE AND LENGTH IN THE DISCRETE CASE

As in the continuous case, the distance between two sequences is defined as the minimum length of any time-warping between the two sequences. Thus the only question is how to form the length of a discrete time-warping $(\mathbf{i}_0, \mathbf{j}_0)$ by analogy with the length of a continuous time-warping. We shall explore several ways of making this analogy. In one approach, the infinitesimal intervals such as $d\mathbf{u}_0(t) = \mathbf{u}_0'(t)\, dt$ correspond to intervals from one sampling point to another, such as $[u_{i-1}, u_i]$. In another approach, each infinitesimal interval corresponds to an interval that surrounds one sampling point, so that each u_i is near the center of its interval. We shall explore several versions of the first approach, and one version of the second approach. It is not clear whether or not the differences among these versions have any substantive importance, but in some cases they do have computational importance. Throughout this section, we assume that sequences **a** and **b** have m and n points, respectively, and are drawn from trajectories $\mathbf{a}(u)$ and $\mathbf{b}(v)$ extending over the time intervals $[0, U]$ and $[0, V]$, respectively. We shall sometimes assume that the sampling times $u_1, \ldots, u_m$ and $v_1, \ldots, v_n$ are regularly and identically spaced, i.e., that $u_i - u_{i-1} = \tau$ and $v_j - v_{j-1} = \tau$ for all i and j.

We start with the first approach. For the time being, we assume that $u_m = U$ and we introduce nonsampling points $u_0 = 0$ and $v_0 = 0$, so that the first and last intervals for $\mathbf{a}(u)$ are $[u_0 = 0, u_1]$ and $[u_{m-1}, u_m = U]$, and similarly for $\mathbf{b}(v)$. To form the analogy, we use the following correspondence for $\mathbf{a}(u)$, and extend it in the obvious way to $\mathbf{b}(v)$:

$$\begin{aligned}
dt &\leftrightarrow 1 = h - (h-1) \\
[u - du,\ u] &\leftrightarrow [u_{\mathbf{i}_0(h-1)},\ u_{\mathbf{i}_0(h)}] \\
d\mathbf{u}_0'(t) = \mathbf{u}_0'(t)\, dt &\leftrightarrow \Delta u_{\mathbf{i}_0(h)} = u_{\mathbf{i}_0(h)} - u_{\mathbf{i}_0(h-1)} \\
\int_0^T [\ldots]\, \mathbf{u}_0'(t)\, dt &\leftrightarrow \sum_{h=1}^{H} [\ldots]\, \Delta u_{\mathbf{i}_0(h)}
\end{aligned}$$

(As a check on the validity of the correspondence, we can apply both the integral and the summation to the function that is identically equal to 1. We obtain $\mathbf{u}_0(T) - \mathbf{u}_0(0) = U$ for the integral and $u_m - u_0 = U$ for the sum.) To complete the analogy, we decide that in the summation we will evaluate the summand $[\ldots]$ using the sampling point at the end rather than the beginning of

its interval, i.e., we will use $u = u_{\mathbf{i}_0(h)}$ rather than $u = u_{\mathbf{i}_0(h-1)}$ in connection with the hth term of the summation. (The opposite decision would not be tenable, because it would involve use of $u = u_0$ when $h = 1$, which is not a sampling point.) Then letting

$$w(h) \equiv_{\text{def}} w[a_{\mathbf{i}_0(h)}, b_{\mathbf{j}_0(h)}],$$

the definition above for length of a continuous time-warping corresponds to the following, which is our first definition for length of a discrete time-warping:

$$\bar{d}(\mathbf{i}_0, \mathbf{j}_0) = \sum_{h=1}^{H} w(h) \frac{[\Delta u_{\mathbf{i}_0(h)} + \Delta v_{\mathbf{j}_0(h)}]}{2}.$$

If two sequences are regularly spaced, as described above, then $\Delta u_{\mathbf{i}(h)} = \tau \Delta \mathbf{i}_0(h)$ and the preceding formula reduces to

$$\bar{d}(\mathbf{i}_0, \mathbf{j}_0) = \tau \sum_{h=1}^{H} w(h) \frac{[\Delta \mathbf{i}_0(h) + \Delta \mathbf{j}_0(h)]}{2}.$$

For the first continuity constraint, the expression in brackets is 1 or 2 or 1 depending on which case occurs, so

$$\bar{d}(\mathbf{i}_0, \mathbf{j}_0) = \tau \sum_{h=1}^{H} z_h w(h),$$

where $z_h = 1$ for a diagonal step and $\frac{1}{2}$ for a vertical or horizontal step. We shall refer to the z_h as the *time weights,* though, properly speaking, it is the products $z_h\tau$ that are the true time weights. The values of z_h are illustrated for this formula in Case 1 of Fig. 8. This length formula is essentially identical to one that is well known in the time-warping literature, and the minimum-length time-warping can be calculated by standard methods. In particular, if D_{ij} = distance = minimum length between the incomplete sequences $a_1 \ldots a_i$ and $b_1 \ldots b_j$, then using recursion to find the values of D_{ij} is the main part of the calculation. Using

$$w(i, j) \equiv_{\text{def}} w[a_i, b_j],$$

the necessary recurrence equation is

$$D_{ij} = \min \begin{cases} D_{i-1,\,j} & + \frac{1}{2}\tau w(i, j), \\ D_{i-1,\,j-1} & + \tau w(i, j), \\ D_{i,\,j-1} & + \frac{1}{2}\tau w(i, j), \end{cases}$$

which can be evaluated recursively using the computational array of Fig. 5.

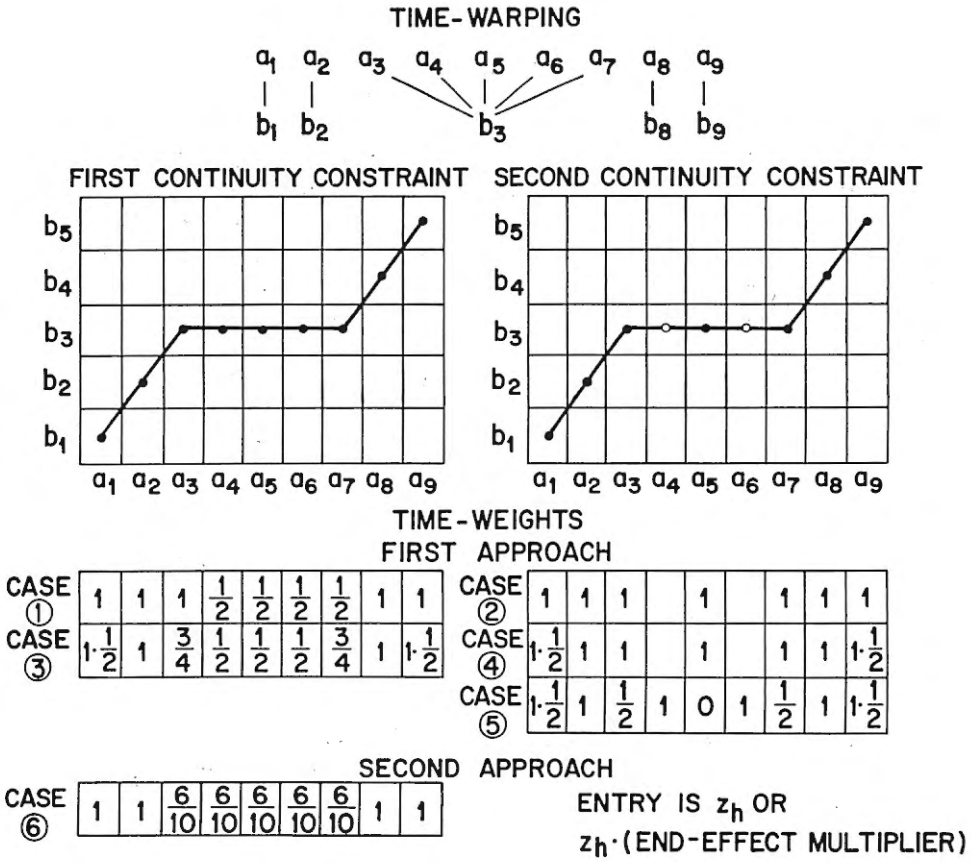


Figure 8. Illustration of time-weights for a given time-warping.

For the second continuity constraint, the expression in brackets is always 2, so the time weights $z_h = 1$ always (see Case 2 of Fig. 8). Using this yields

$$\bar{d}(\mathbf{i}_0, \mathbf{j}_0) = \tau \sum_{h=1}^{H} w(h).$$

This length formula is the same as that used in Chapter 5 by Hunt, Lennig, and Mermelstein. The minimization can easily be carried out by methods virtually identical to the standard ones, as illustrated in that chapter (of course, only half of the cells of the computational array are used, namely cells (i, j) with $i + j$ even). In particular, the recurrence equation is

$$D_{ij} = \min \begin{cases} D_{i-2,\,j} & + \tau w(i, j), \\ D_{i-1,\,j-1} & + \tau w(i, j), \\ D_{i,\,j-2} & + \tau w(i, j). \end{cases}$$

Suppose we proceed as above, but with one change. Instead of using $w(h)$, which means evaluating the summand at the end of the interval, suppose we use $[w(h-1) + w(h)]/2$, which means evaluating at both ends and taking the average. This requires a slight change of convention to avoid evaluating the summand at u_0 and v_0, which are not sampling points. Thus we set $u_1 = v_1 = 0$ (so the first interval for $\mathbf{a}(u)$ is $[u_1, u_2]$). This leads to

$$\bar{d}(\mathbf{i}_0, \mathbf{j}_0) = \tau \sum_{h=2}^{H} \frac{w(h-1) + w(h)}{2} \cdot \frac{\Delta \mathbf{i}_0(h) + \Delta \mathbf{j}_0(h)}{2}.$$

For the first continuity constraint, we find that

$$\bar{d}(\mathbf{i}_0, \mathbf{j}_0) = \tau \left\{ \tfrac{1}{2} z_1 w(1) + \sum_{h=2}^{H-1} z_h w(h) + \tfrac{1}{2} z_H w(H) \right\},$$

where the time weights

$$z_h = \tfrac{1}{2} \quad \text{or} \quad \tfrac{3}{4} \quad \text{or} \quad 1$$

(see Case 3 of Fig. 8) depending on the steps which end with h and start with h (with a special rule for $h = 1$ and $h = H$). Again, the minimization can be carried out as usual. An appropriate equation is

$$D_{ij} = \min \begin{cases} D_{i-1,\,j} & + \tfrac{1}{4}\tau[w(i-1, j) + w(i, j)], \\ D_{i-1,\,j-1} & + \tfrac{1}{2}\tau[w(i-1, j-1) + w(i, j)], \\ D_{i,\,j-1} & + \tfrac{1}{4}\tau[w(i, j-1) + w(i, j)]. \end{cases}$$

For the second continuity constraint, we find that

$$\bar{d}(\mathbf{i}_0, \mathbf{j}_0) = \tau \left\{ \tfrac{1}{2} w(1) + \sum_{h=2}^{H-1} w(h) + \tfrac{1}{2} w(H) \right\},$$

which is reminiscent of the trapezoid rule for integration. Here $z_h = 1$ always (see Case 4 of Fig. 8). A recurrence equation for the minimization is

$$D_{ij} = \min \begin{cases} D_{i-2,\,j} & + \ \tfrac{1}{2}\tau[w(i-2,\,j) + w(i,\,j)], \\ D_{i-1,\,j-1} & + \ \tfrac{1}{2}\tau[w(i-1,\,j-1) + w(i,\,j)], \\ D_{i,\,j-2} & + \ \tfrac{1}{2}\tau[w(i,\,j-2) + w(i,\,j)]. \end{cases}$$

As an interesting side note, when using the second continuity constraint it is possible to use $w(h - \frac{1}{2})$ in place of $[w(h-1) + w(h)]/2$ for a vertical or horizontal step, because such a step moves two places along one of the sequences. Using such evaluation where possible leads to still another length formula, which we omit (but see Case 5 of Fig. 8), and the following recurrence equation:

$$D_{ij} = \min \begin{cases} D_{i-2,\,j} & + \ \tau w(i-1,\,j), \\ D_{i-1,\,j-1} & + \ \tfrac{1}{2}\tau[w(i-1,\,j-1) + w(i,\,j)] \\ D_{i,\,j-2} & + \ \tau w(i,\,j-1). \end{cases}$$

Consider the second approach to forming the intervals. We assume that sequences **a** and **b** were formed by dividing the trajectories into m and n pieces lasting time τ each and placing a sample point centrally in each interval. Then the definition of length for a continuous time-warping corresponds to

$$\bar{d}(\mathbf{i}_0,\,\mathbf{j}_0) = \sum_{h=1}^{H} w(h) z_h \tau$$

if we choose $z_h\tau$ analogous to $(\mathbf{u}_0'(t)dt + \mathbf{v}_0'(t)dt)/2$. Now $\mathbf{u}_0'(t)dt$ indicates time spent in trajectory $\mathbf{a}(u)$, and similarly for $\mathbf{v}_0'(t)dt$. Thus $z_h\tau$ should be the average of the time spent corresponding to h in the sequence **a** and the time spent corresponding to h in sequence **b**. One way to give this specific meaning relies on the constraint (see above) forbidding the presence of an "N"-shaped configuration in the discrete time-warping diagram. With this constraint, the diagram divides naturally into connected component groups, which are of three types (and see Case 6 of Figure 8):

i) A single a_i joined to a single b_j. This group contains one value of h, and it corresponds to time τ in each sequence, so the average is τ and we set $z_h = 1$.

ii) A single a_i joined to k terms from **b** (with $k \geqq 2$). This group contains k values of h. It corresponds to time τ in the **a** sequence, and to time $k\tau$ in the **b** sequence. Taking the average, and dividing the amount of time evenly into k parts, we get $\tau(k+1)/2k$, so we set $z_h = (k+1)/2k$ for each of the k time-weights in the group.
iii) A single b_j joined to k terms from **a** (with $k \geqq 2$). In a similar manner, we find that $z_h = (k+1)/2k$ for each of the k time-weights in the group.

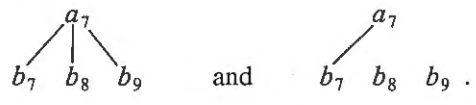
(Note that if we drop the requirement $k \geqq 2$, then the latter two cases are consistent with the first one.) The recurrence equation for this definition of length is computationally slower than the recurrence equations above:

$$D_{ij} = \min \begin{cases} \min\limits_{2 \leqq i_1 < i} \left[D_{i-i_1, j-1} + \tau \dfrac{i_1 + 1}{2i_1} \sum\limits_{i_2=1}^{i_1} w(i + 1 - i_2, j) \right], \\ D_{i-1, j-1} + d(i, j), \\ \min\limits_{2 \leqq j_1 < j} \left[D_{i-1, j-j_1} + \tau \dfrac{j_1 + 1}{2j_1} \sum\limits_{j_2=1}^{j_1} w(i, j + 1 - j_2) \right]. \end{cases}$$

6. HOW COMPRESSION–EXPANSION DIFFERS FROM DELETION–INSERTION

We have already noted one difference between compression–expansion and deletion–insertion in Sec. 3, when we contrasted the concepts of linking and trace. There is, however, a more important difference.

Consider the following bit from a time-warping diagram, and a very similar bit from a trace diagram:



The former expands a_7 to match $b_7b_8b_9$; the latter inserts b_8 and b_9. By redescribing compression–expansion systematically in this manner, it is converted into deletion–insertion, and the time-warping problem can be thought of as the deletion–insertion problem. It is through this relationship that the two sequence-comparison problems have often been considered the same.

Despite this conversion, the problems are not the same. The difference between expansion and insertion lies in the costs that we wish to assign to these operations. In the trace, the cost assigned to the substitution of a_7 by b_7 is treated very differently from the cost of the insertions of b_8 and b_9. The cost of the substitution will be small if b_7 equals or resembles a_7, and will be larger otherwise. By contrast, there is no reason for the weight of the insertions to be small if b_8 or b_9 equals or resembles a_7; nor is there even any reason to select a_7 for the comparison over, say, a_8. The ultimate reason for this treatment of the costs is the basic physical processes we have in mind, namely, substitution or modification of a_7 to yield b_7, but insertion of a new element rather than modification of an existing one to yield b_8 and b_9.

On the other hand, in time-warping the costs assigned to each of the three comparisons (a_7 with each of b_7, b_8, b_9) are all treated in similar or identical fashion, and for each of them the cost should be smaller if b_j equals or resembles a_7. The ultimate reason for this treatment of the weights is again the basic physical process we have in mind, namely, that the trajectory moves more slowly through the region around a_7 on the second replication, so this region is represented by three points instead of one, and the difference between a_7 and b_j is due to additive random error.

7. COMBINING DELETION–INSERTION AND COMPRESSION–EXPANSION IN A SINGLE METHOD

One problem in applying time-warping to speech processing is that speech utterances may differ not only by time-distortion and additive random error but also by interpolated or deleted sounds. This can happen for a variety of reasons: extraneous sounds from the ambient environment (door slamming, footsteps, etc.), speaker-generated nonspeech sounds (lip smacks, coughs, breath noises, etc.), and more or less full pronunciation of a word (the dictionary pronunciation of "probably" may be reduced to "prob'ly" or even "pro'lly," the dictionary pronunciation "offen" may be expanded to the spelling pronunciation "often," etc.). One step towards dealing with such additional difficulties is to perform the comparison in a way that allows for deletion–insertion as well as compression–expansion. (In the case of an extraneous sound that does not delay the normal speech but merely conceals a bit of it, deletion–insertion permits the concealed bit to be deleted and the extraneous sound to be inserted, which is a more realistic and perhaps more desirable explanation than that permitted by additive random error.) Although this appears not to have been done before, it can be done in a simple and computationally tractable manner. Of course, there are even more alternative versions in this case than there are when only compression–expansion is permitted, but we restrict our discussion

here. While these simple versions may not in themselves be adequate to handle the kind of difficulties referred to, they do indicate one approach to dealing with them. A more sophisticated approach is briefly mentioned.

The time warping is defined by the functions $\mathbf{i}_0(h)$ and $\mathbf{j}_0(h)$ as before, and we use the first continuity constraint,

$$(\Delta\mathbf{i}_0, \Delta\mathbf{j}_0) = \begin{cases} (1, 0) & \text{or} \\ (1, 1) & \text{or} \\ (0, 1), \end{cases}$$

(where $\Delta\mathbf{i}_0(h) = \mathbf{i}_0(h) - \mathbf{i}_0(h-1)$), but the case (1, 0) can refer either to compression as before or, instead, to deletion of $a_{\mathbf{i}_0(h)}$; likewise, the case (0, 1) can refer either to expansion or to insertion of $b_{\mathbf{j}_0(h)}$. We introduce deletion and insertion weights $w_{\text{del}}[a_i]$ and $w_{\text{ins}}[b_j]$ in addition to the substitution weights $w[a_i, b_j]$. We think of these as weights *per unit time*, so that the cost for a deletion of a_i over an interval of τ time units is $\tau w_{\text{del}}[a_i]$.

The simplest recurrence that can be used is

$$D_{ij} = \min \begin{cases} D_{i-1,\,j} & + \ \tfrac{1}{2}\tau w[a_i, b_j], \\ D_{i-1,\,j} & + \ \tau w_{\text{del}}[a_i], \\ D_{i-1,\,j-1} & + \ \tau w[a_i, b_j], \\ D_{i,\,j-1} & + \ \tau w_{\text{ins}}[b_j], \\ D_{i,\,j-1} & + \ \tfrac{1}{2}\tau w[a_i, b_j]. \end{cases}$$

However, the justification for time weights of $\frac{1}{2}$ for compression and expansion steps does not seem valid if such a step immediately follows a deletion or an insertion step. For that matter, it is probably more realistic to forbid a compression or expansion step immediately following a deletion or insertion step, and doing so improves the time-reversal symmetry of the linking concept. If we choose to do this, two coupled recurrences may be used. Let D_{ij}^{di} ("di" for deletion–insertion) be the length of the shortest possible time-warping between $a_1 \ldots a_i$ and $b_1 \ldots b_j$ that ends with a deletion or an insertion, and let d_{ij}^{o} ("o" for other) be the length of the best possible time-warping that ends in some other step. Then

$$D_{ij}^{\text{di}} = \min \begin{cases} \min\,(D_{i-1,\,j}^{\text{d i}}, D_{i-1,\,j}^{\text{o}}) + & \tau w_{\text{del}}[a_i], \\ \min\,(D_{i,\,j-1}^{\text{d i}}, D_{i,\,j-1}^{\text{o}}) \ + & \tau w_{\text{ins}}[b_j], \end{cases}$$

$$D^{\circ}_{ij} = \min \begin{cases} D^{\circ}_{i-1,\,j} & + \tfrac{1}{2}\tau w[a_i,\, b_j], \\ D^{\circ}_{i,\,j-1} & + \tfrac{1}{2}\tau w[a_i,\, b_j], \\ \min\,(D^{\mathrm{di}}_{i-1,\,j-1},\, D^{\circ}_{i-1,\,j-1}) & + \tau w[a_i,\, b_j], \end{cases}$$

are the desired recurrences, and $\min(D^{\mathrm{di}}_{mn},\, D^{\circ}_{mn})$ gives the distance between **a** and **b**.

It is possible to extend these versions in a much more general approach that offers the possibility of considerable computational saving, though we mention this only briefly. By making the template utterance a network and generalizing the comparison problem as described in the section on Directed Networks in Chapter 10, we can treat normal speech-sound deletion–insertion (e.g., "probably" to "prob'ly") as an alternative path in a network, rather than as a long series of deletions or insertions of individual sequence elements. One computational advantage of this comes from the fact that deletion-insertion need not be considered as a possibility at every element in the sequence (which is, in any case, unrealistic for speech sounds), but only at a few specified places. It is still feasible and perhaps helpful to permit arbitrary deletion–insertion to handle extraneous sounds when using this network approach, because the weights used for the two different types of deletion–insertion may be quite different. We omit a more concrete description, since it would take us too far afield.

8. INTERPOLATION BETWEEN THE SAMPLING POINTS

The sampling procedure that converts trajectories $\mathbf{a}(u)$ and $\mathbf{b}(v)$ into sequences $\mathbf{a} = a_1 \ldots a_m$ and $\mathbf{b} = b_2 \ldots b_n$ can sometimes cause a problem when the curves swept out by the trajectories are very close together. (This problem has been encountered in x-ray pellet-tracking measurements of tongue and jaw movements during speech.) Suppose, as in Fig. 9, that the distance from one curve to the other in some region is small compared to the typical distances between successive sampling points (such as a_i to a_{i+1}, and b_j to b_{j+1}) in the same region. If the sampling in the two trajectories happens by chance to be nearly in phase, as in Fig. 9(a), then the optimum time-warping between the sequences may give a satisfactory description of the relationship between the two underlying trajectories. If, however, the sampling happens to be substantially out of phase, as in Fig. 9(b) or Fig. 9(c), then the values like $w[a_i, b_j]$ that enter into the length of the time-warping are much larger than the distance between the curves. In this case, the discrete time-warping gives an unduly pessimistic result. In addition, the correspondence between the trajectories is substantially less accurate than is possible by more sophisticated analysis of the same data.

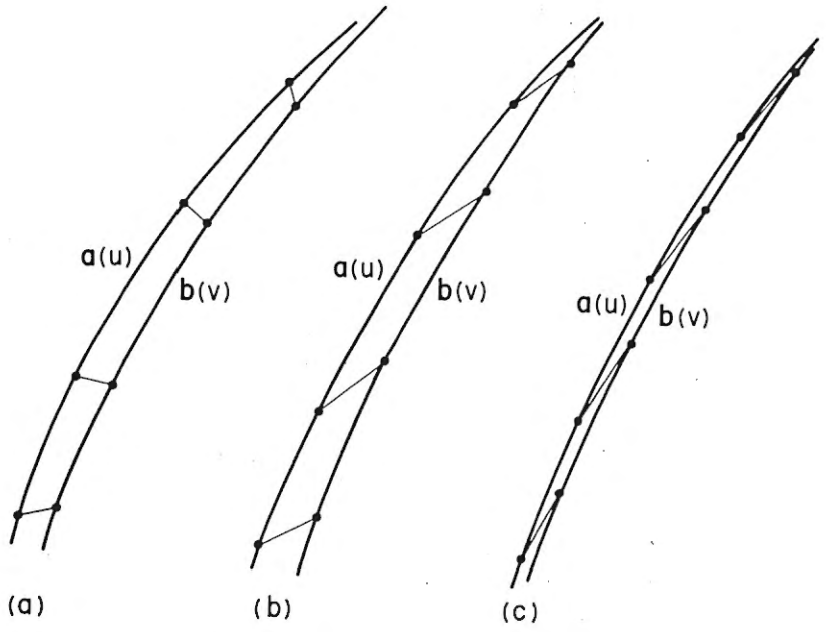


Figure 9. Close trajectories with sampling in or out of phase.

To overcome these problems, we have devised a new type of discrete time-warping, which we call *interpolation time-warping.* (A very different method for a somewhat similar purpose may be found in Burr (1979).) It makes use of two polygonal paths in feature space: the **a**-path that connects the a_i in order, and the **b**-path that connects the b_j in order. Each a_i is matched not to some b_j but instead to some point on the **b**-path, and each b_j is matched to some point on the **a**-path.

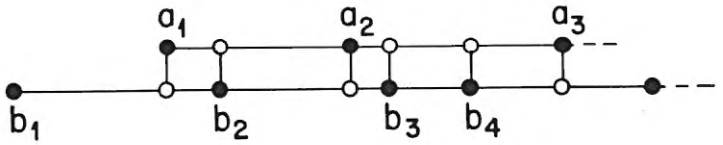
To describe an interpolation time-warping, we can use a diagram like Fig. 10(a), whose meaning is indicated by Fig. 10(b). Each $r(h)$ is a number between 0 and 1, and the notation (a_i, r) indicates an *interpolation point* on the segment from a_i to a_{i+1}. Specifically, (a_i, r) is the point that is r of the way from a_i to a_{i+1}, that is,

$$(a_i, r) \equiv_{\text{def}} (1 - r)a_i + ra_{i+1}.$$

Diagram 10(a) shows that a_1 is matched to the interpolation point $(b_1, r(1))$ between b_1 and b_2, and that b_2 is matched to an interpolation point between a_1 and a_2, etc. If the sampling rate is the same in the two trajectories, sampling points and interpolation points will tend to alternate along each row of the

$$\begin{bmatrix} a_1 & (a_1, r(2)) & a_2 & (a_2, r(4)) & (a_2, r(5)) & a_3 & \dots \\ (b_1, r(1)) & b_2 & (b_2, r(3)) & b_3 & b_4 & (b_4, r(6)) & \dots \end{bmatrix}$$

(a)



(b)

Figure 10. Interpolation time-warping.

diagram, but alternation need not occur, as illustrated by adjacent sampling points b_3 and b_4, which are both matched to interpolation points in the same segment.

More formally, an *interpolation time-warping between* $\mathbf{a} = a_1 \dots a_m$ and $\mathbf{b} = b_1 \dots b_n$ consists of functions $\mathbf{I}_0(h)$, $\mathbf{J}_0(h)$, $\mathbf{i}_0(h)$, $\mathbf{j}_0(h)$, and $\mathbf{r}(h)$. Each of them is defined for $h = 1$ to H, where $H = m + n - 2$. For each h, either

i) $\mathbf{I}_0(h)$ is the sampling point $a_{\mathbf{i}_0(h)}$ and $\mathbf{J}_0(h)$ is the interpolation point $(b_{\mathbf{j}_0(h)}, \mathbf{r}(h))$, or
ii) $\mathbf{I}_0(h)$ is the interpolation point $(a_{\mathbf{i}_0(h)}, \mathbf{r}(h))$ and $\mathbf{J}_0(h)$ is the sampling point $b_{\mathbf{j}_0(h)}$.

The functions $\mathbf{I}_0$ and $\mathbf{i}_0$ are required to satisfy the following continuity constraint, and $\mathbf{J}_0$ and $\mathbf{j}_0$ are required to satisfy a similar one.

- **a)** If $\mathbf{I}_0(h)$ and $\mathbf{I}_0(h+1)$ are both sampling points, then they are adjacent in $\mathbf{a}$, that is, $\mathbf{i}_0(h) + 1 = \mathbf{i}_0(h+1)$.
- **b)** If $\mathbf{I}_0(h)$ is a sampling point and $\mathbf{I}_0(h+1)$ is an interpolation point, then the sampling point is the beginning point of the segment containing the interpolation point, that is, $\mathbf{i}_0(h) = \mathbf{i}_0(h+1)$.
- **c)** If $\mathbf{I}_0(h)$ is an interpolation point and $\mathbf{I}_0(h+1)$ is a sampling point, then

the line segment containing the interpolation point ends at the sampling point, that is, $\mathbf{i}_0(h) + 1 = \mathbf{i}_0(h+1)$.

d) If $\mathbf{I}_0(h)$ and $\mathbf{I}_0(h+1)$ are both interpolation points, they belong to the same line segment, that is, $\mathbf{i}_0(h) = \mathbf{i}_0(h+1)$.

To insure that the time-warping covers the entire sequences, we can use the following constraint:

$$\mathbf{i}_0(1) = \mathbf{j}_0(1) = 1$$

and

$$\begin{cases} \mathbf{i}_0(H) = m & \text{and } \mathbf{j}_0(H) = n-1, \quad \text{if } \mathbf{I}_0(H) \text{ is a sampling point,} \\ \mathbf{i}_0(H) = m-1, & \text{and } \mathbf{j}_0(H) = n, \quad \text{if } \mathbf{J}_0(H) \text{ is a sampling point.} \end{cases}$$

If we assume that the sampling interval is the constant 2τ for both trajectories, then it is very simple to define length appropriately for an interpolation time-warping:

$$d(\mathbf{I}_0, \mathbf{J}_0, \mathbf{i}_0, \mathbf{j}_0, \mathbf{r}) \equiv \tau \sum_{h=1}^{H} w[\mathbf{I}_0(h), \mathbf{J}_0(h)].$$

The reason this is appropriate is that every h can be considered to correspond to time 2τ in the trajectory where a sampling point is used, and to time 0 in the trajectory where an interpolation point is used, so h corresponds to the average, τ. Although generalizing the standard recursion to handle this case involves some novel features, the resulting recurrence equation is quite simple and easy to calculate. Let A_{ij} be the set of incomplete interpolation time-warpings that start from the beginning of **a** and **b**, that is, $\mathbf{i}_0(1) = \mathbf{j}_0(1) = 1$, and that end at (i, j), that is, $\mathbf{i}_0(h_{\text{last}}) = i$ and $\mathbf{j}_0(h_{\text{last}}) = j$. (Here $h_{\text{last}} = i + j - 1$.) In other words, the time-warpings in A_{ij} are incomplete because they end at (i, j), which means that a_i is matched to an interpolation point between b_j and b_{j+1}, or b_j is matched to an interpolation point between a_i and a_{i+1}. Let D_{ij} be the minimum length of any time-warping in A_{ij}. Then the recurrence for D_{ij} is

$$D_{ij} = \min \begin{cases} D_{i-1,\,j} + \min\limits_{0 \le r \le 1} \tau w[a_i, (b_j, r)], \\ D_{i,\,j-1} + \min\limits_{0 \le r \le 1} \tau w[(a_i, r), b_j], \end{cases}$$

and the minimum length of any complete interpolation time-warping is given by

$$d(\mathbf{a}, \mathbf{b}) = D_{m-1,\,n-1} + \min \begin{cases} \min\limits_{0 \le r \le 1} \tau w[a_m, (b_{n-1}, r)], \\ \min\limits_{0 \le r \le 1} \tau w[(a_{m-1}, r), b_n]. \end{cases}$$

(Note the differences in subscript pattern between this formula and the preceding one.) Each minimization over r corresponds to finding the point on a given line segment that is nearest to a given point, and this is easy to calculate. For example, the first minimization over r in the formula for D_{ij} corresponds to finding the point on the segment from b_j to b_{j+1} that is nearest to a_i. To find it, project a_i perpendicularly onto the complete line between b_j and b_{j+1}. If the projection is between b_j and b_{j+1}, it is the desired point. If the projection is beyond b_j, then b_j is the desired point, and if the projection is beyond b_{j+1}, then b_{j+1} is the desired point. In algebraic terms, this can be written as follows,

$$\bar{r} = \frac{(a_i - b_j) \cdot (b_{j+1} - b_j)}{(b_{j+1} - b_j) \cdot (b_{j+1} - b_j)},$$

$$r = \begin{cases} 1, & \text{if } \bar{r} > 1, \\ 0, & \text{if } \bar{r} < 0, \\ \bar{r}, & \text{otherwise,} \end{cases}$$

where the dot indicates the scalar product of two vectors.

We can generalize interpolation time-warpings to permit insertion and deletion in a way that is appropriate for use in speech processing (e.g., to get a good comparison between a precise pronunciation of "twenty" and the slurred pronunciation "twenny," by permitting deletion of "t"). The recurrence equation is straightforward, but the algorithm requires a three-dimensional array and time proportional to n^3 instead of n^2.

9. AVERAGE OF TWO TRAJECTORIES

It is sometimes useful to take the "average" of several trajectories, as illustrated in Rabiner and Wilpon (1979, 1980). The prime application occurs in speech processing, where several utterances of a single word are combined into a single average utterance to provide a "template" for use in word recognition. The Rabiner–Wilpon method has the advantage of being relatively simple, and of permitting a simple extension to the average of many trajectories (with one of them playing a special master role). It treats the trajectories in an asymmetric manner, however, and in this paper we are interested in developing a fully symmetric method, for reasons described earlier. We will give a natural symmetric definition for the weighted average of two trajectories with respect to a given time-warping between them. Of course, the optimum time-warping would normally be used. In principle, our definition could be extended to averaging N trajectories, but such an extension would rest on a simultaneous time-warping of N trajectories. We do not follow this approach, in part because

of the great computation time such methods require. Instead, to combine N trajectories into a single average, $N-1$ repetitions of the two-trajectory average may be used, each with respect to the optimum time-warping between the two trajectories involved. For example, various pairs may be combined, then some of the resulting trajectories may be combined, either with each other or with original trajectories that have not yet been combined, and so on. When combining two trajectories that represent k_1 and k_2 original trajectories, respectively, presumably we would use weights $k_1/(k_1+k_2)$ and $k_2/(k_1+k_2)$. While the final average would not, unfortunately, be independent of the order of combination, it would probably not be very sensitive to the order, in realistic applications. Incidentally, the combining process described bears a strong relationship to widely used methods of clustering known as "pair-group" methods, and the rules used in clustering to determine the order of combination are probably quite suitable for use here also.

Now assume we are given the trajectories $\mathbf{a}(u)$ and $\mathbf{b}(v)$, and weights p and q with $p+q=1, p \geq 0, q \geq 0$. Also assume that we are given a time-warping $(\mathbf{u}_0, \mathbf{v}_0)$ between the trajectories. Presumably, this would usually be the optimum time-warping, though the following discussion does not rely on that assumption. Suppose u and v are linked, so that $\mathbf{a}(u)$ and $\mathbf{b}(v)$ are corresponding points in the two trajectories. The weighted average of these points is

$$p\mathbf{a}(u) + q\mathbf{b}(v).$$

Obviously the weighted-average trajectory should run along the curve formed by all such points.

What has not been so clearly set forth in the literature is the time pattern that should be used with this curve. We propose that the time assigned to the point shown above should be the weighted average of the two times involved, u and v, that is,

$$w = pu + qv$$

is the time that should be assigned to the point shown above. (It may appear that the time pattern chosen for the average is not important, because the distances we use are deliberately chosen to be insensitive to time pattern. However, it is in fact important, for two reasons. First, a time pattern is needed to use the procedures we discuss. Second, other ways of using the average trajectory, not discussed in this chapter, are sensitive to time pattern.)

This can all be wrapped up into one succinct definition, as follows. The *weighted average* $\mathbf{c}(w)$ of trajectories $\mathbf{a}(u)$ and $\mathbf{b}(v)$, with respect to the time-warping $(\mathbf{u}_0, \mathbf{v}_0)$, using nonnegative weights p and q that sum to 1, is defined by

$$\mathbf{c}(\mathbf{w}_0(t)) = p\mathbf{a}(\mathbf{u}_0(t)) + q\mathbf{b}(\mathbf{v}_0(t))$$

where

$$\mathbf{w}_0(t) = p\mathbf{u}_0(t) + q\mathbf{v}_0(t).$$

If equivalent time-warpings $(\mathbf{u}_0, \mathbf{v}_0)$ and $(\mathbf{u}_1, \mathbf{v}_1)$ between $\mathbf{a}(u)$ and $\mathbf{b}(v)$ are each used to form the average, then it is easy to show that the two resulting average trajectories are the same.

10. AVERAGE OF TWO SEQUENCES

In the previous section, a reason for averaging trajectories was explained, but in practice, of course, it is *sequences* that are averaged. In this section, we define the average of two sequences with respect to a time-warping. As above, there are many ways to make the analogy with the continuous definition, and we give two alternative definitions. One difference between our definitions and the earlier definitions due to Rabiner and Wilpon (1979, 1980) is that we treat the two sequences in a fully symmetric manner.

Suppose that we are given two sequences $\mathbf{a} = a_1 \ldots a_m$ and $\mathbf{b} = b_1 \ldots b_n$ with sampling times u_i and v_j, and weights p and q with $p + q = 1, p \geqq 0, q \geqq 0$. Also assume that we are given a time-warping $(\mathbf{i}_0, \mathbf{j}_0)$ between the sequences, where $\mathbf{i}_0(h)$ and $\mathbf{j}_0(h)$ are defined for $h = 1$ to H. Presumably, the optimum time-warping would normally be used, but the following discussion is valid for any time-warping. Suppose i and j are linked by h (that is, $i = \mathbf{i}_0(h), j = \mathbf{j}_0(h)$), so that a_i and b_j are corresponding points in the two sequences. The weighted average of these points is

$$c_h = pa_i + qb_j,$$

which corresponds to h. Let the corresponding sampling time be defined by

$$w_h = pu_i + qv_j.$$

Our first definition of the average sequence is simply $\mathbf{c} = c_1 \ldots c_H$.

If $\mathbf{a}$ and $\mathbf{b}$ were both formed by sampling at constant time intervals τ, then we might want the average to have the same property. Our second definition achieves this, though there are many alternative versions of it, and we shall not spell out the details for any one of them. First we put a polygonal path (or more generally, a spline) through the points c_h. We label the points with w_h, and interpolate along the path to find points at the appropriate sampling times. The points yielded by this process constitute our second definition for the average.

REFERENCES

Bridle, J. S., and Brown, M. D., Connected word recognition using whole-word templates, *Proceedings of the Institute for Acoustics*, pages 25–28 (1979).

Burr, D. J., A technique for comparing curves, pages 271–277, in *Proceedings of the IEEE Conference on Pattern Recognition and Image Processing, 1979, Chicago,* IEEE: New York (1979).

Burr, D. J., Designing a handwriting reader, pages 715–722, in *International Conference on Pattern Recognition, 5th, 1980, Miami Beach, Florida,* IEEE: New York (1980).

Burr, D. J., Elastic matching of line drawings, *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-3:708–713 (1981).

Dixon, N. R., and Martin, T. B. (eds.), *Automatic speech and speaker recognition,* IEEE Press: New York City (1979).

Fujimura, O., Temporal organization of articulatory movements as a multidimensional phrasal structure, *Phonetica* **38**:66–83 (1981).

Fujimoto, Y., Kadota, S., Hayashi, S., Yamamoto, M., Yajima, S., and Yasuda, M., Recognition of handprinted characters by nonlinear elastic matching, pages 113–119, *International Joint Conference on Pattern Recognition, 3rd 1976, Coronado, California,* IEEE: New York (1976).

Itakura, F., Minimum prediction residual principle applied to speech recognition, *IEEE Transactions on Acoustics, Speech, and Signal Processing,* ASSP-**23**:67–72 (1975).

Levinson, S. E., Structural pattern recognition applied to automatic speech recognition, SCAMP Working Paper No. 13/81, in Proceedings of a Symposium on Acoustics Phonetics and Speech Modeling, June 1981, Volume 2, published by Communications Division, Institute for Defense Analyses, Princeton, N.J. (A.S. House, editor) (1981).

Myers, C., A comparative study of several dynamic time-warping algorithms for speech recognition, Master of Science thesis, Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology, February 1980.

Myers, C. S., and Rabiner, L. R., A dynamic time-warping algorithm for connected word recognition, *IEEE Transactions for Acoustics, Speech, and Signal Processing* ASSP-**29**:284–297 (1981a).

Myers, C. S., and Rabiner, L. R., Connected-digit recognition using a level-building DTW algorithm, *IEEE Transactions on Acoustics, Speech, and Signal Processing* ASSP-**29**:351–363 (1981b).

Myers, C. S., Rabiner, L. R., and Rosenberg, A. E., Performance tradeoffs in dynamic time-warping algorithms for isolated-word recognition, *IEEE Transactions on Acoustics, Speech, and Signal Processing* ASSP-**28**:622–635 (1980).

Rabiner, L. R., Rosenberg, A. E., and Levinson, S. E., Considerations in dynamic time-warping for discrete-word recognition, *IEEE Transactions on Acoustics, Speech, and Signal Processing* ASSP-**26**:575–582 (1978).

Rabiner, L. R., and Schmidt, C. E., Application of dynamic time-warping to connected-digit recognition, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-**28**:337–388 (1980).

Rabiner, L. R., and Wilpon, J. G., Considerations in applying clustering techniques to speaker-independent word recognition, *Journal of the Acoustical Society of America,* **66**(3):663–673 (1979).

Rabiner, L. R., and Wilpon, J. G., A simplified, robust training procedure for speaker-trained, isolated-word recognition systems. *Journal of the Acoustical Society of America,* **68**(5): 1271–1276 (1980).

Reiner, E., and Bayer, F. L., Botulism: A pyrolysis-gas-liquid chromatographic study. *Journal of Chromatographic Science* **16**(12):623–629 (1978).

Reiner, E., and Kubica, G. P., Predictive value of pyrolysis-gas-liquid chromatography in the differentiation of mycobacteria. *American Review of Respiratory Disease* **99**:42–49 (1969).

Reiner, E., Abbey, L. E., Moran, T. F., Papamichalis, P., and Schafer, R. W., Characterization of normal human cells by pyrolysis-gas-chromatography mass spectrometry. *Biomedical Mass Spectrometry* **6**(11):491–498 (1979).

Sakoe, H., and Chiba, S., Dynamic-programming algorithm optimization for spoken word recognition, *IEEE Transactions for Acoustics, Speech, and Signal Processing,* ASSP-**26**:43–49 (1978).

Sakoe, H., and Chiba, S., A dynamic-programming approach to continuous speech recognition, *1971 Proceedings of the International Congress of Acoustics, Budapest, Hungary*, Paper 20 C13 (1971).

Velichko, V. M., and Zagoruyko, N. G., Automatic recognition of 200 words. *International Journal of Man–Machine Studies,* **2**:223–234 (1970).

White, G. M., and Neely, R. B., Speech-recognition experiments with linear prediction, bandpass filtering, and dynamic programming, *IEEE Transactions on Acoustics, Speech, and Signal Processing,* ASSP-**24**:183–188 (1976).

White, G. M., Dynamic programming, the Viterbi algorithm, and low-cost speech recognition, pages 413–417 in *Proceedings of the 1978 IEEE International Conference on Acoustics, Speech, and Signal Processing* (1978).

Yasuhara, M., and Oka, M., Signature-verification experiment based on nonlinear time alignment: feasibility study. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-7:212–216 (1977).