Derivative Dynamic Time Warping

Eamonn J. Keogh[†] and Michael J. Pazzani[‡]

1 Introduction

Time series are a ubiquitous form of data occurring in virtually every scientific discipline. A common task with time series data is comparing one sequence with another. In some domains a very simple distance measure, such as Euclidean distance will suffice. However, it is often the case that two sequences have the approximately the same overall component shapes, but these shapes do not line up in X-axis. Figure 1 shows this with a simple example. In order to find the similarity between such sequences, or as a preprocessing step before averaging them, we must "warp" the time axis of one (or both) sequences to achieve a better alignment. Dynamic time warping (DTW), is a technique for efficiently achieving this warping. In addition to data mining (Keogh & Pazzani 2000, Yi et. al. 1998, Berndt & Clifford 1994), DTW has been used in gesture recognition (Gavrila & Davis 1995), robotics (Schmill et. al 1999), speech processing (Rabiner & Juang 1993), manufacturing (Gollmer & Posten 1995) and medicine (Caiani et. al 1998).

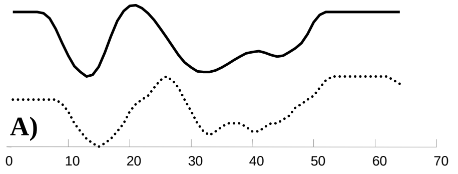

Figure 1: An example of the utility of dynamic time warping. **A)** Two sequences that represent the Y-axis position of an individual's hand while signing the word "pen" in Sign Language. The sequences were recorded on two separate days. Note that while the sequences have an overall similar shape, they are not aligned in the time axis. A distance measure that assumes the ith point on one sequence is aligned with ith point on the other will produce a pessimistic dissimilarity. **B)** DTW can efficiently find an alignment between the two sequences that allows a more sophisticated distance measure to be calculated.

[†] Department of Information and Computer Science University of California, Irvine, California 92697 USA *eamonn@ics.uci.edu*
[‡] *pazzani@ics.uci.edu*

Although DTW has been successfully used in many domains, it can produce pathological results. The crucial observation is that the algorithm may try to explain variability in the Y-axis by warping the X-axis. This can lead to unintuitive alignments where a single point on one time series maps onto a large subsection of another time series. We call examples of this undesirable behavior "singularities". A variety of ad-hoc measures have been proposed to deal with singularities. All of these approaches essentially constrain the possible warpings allowed. However they suffer from the drawback that they may prevent the "correct" warping from being found.

In simulated cases, the correct warping can be known by warping a time series and attempting to recover the original (see Section 4). In naturally occurring cases we take "correct" to mean intuitively obvious "feature to feature" alignment as in Figure 2.B.

An additional problem with DTW is that the algorithm may fail to find obvious, natural alignments in two sequences simply because a feature (i.e peak, valley, inflection point, plateau etc.) in one sequence is slightly higher or lower than its corresponding feature in the other sequence. Figure 2 illustrates this problem.

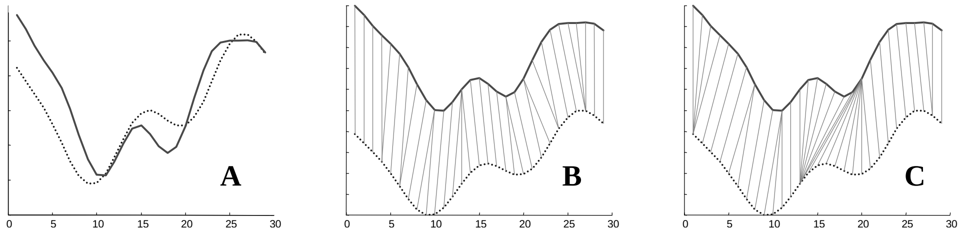

Figure 2 : A) Two synthetic signals (with the same mean and variance). **B**) The natural "feature to feature" alignment. **C**) The alignment produced by dynamic time warping. Note that DTW failed to align the two central peaks because they are slightly separated in the Y-axis

In this paper we address both these problems by introducing a modification of DTW. The crucial difference is in the features we consider when attempting to find the correct warping. Rather than use the raw data, we consider only the (estimated) local derivatives of the data.

The rest of the paper is organized as follows. Section 2 contains a review of the classic DTW algorithm, including the various techniques suggested to prevent singularities. In Section 3 we introduce and demonstrate our extension, which we call Derivative Dynamic Time Warping (DDTW). Section 4 contains experimental results, and in Section 5 we offer conclusions and discuss possible directions for future work.

2 The classic dynamic time warping algorithm

Suppose we have two time series Q and C, of length n and m respectively, where:

$$Q = q_1,q_2,\ldots,q_i,\ldots,q_n \tag{1}$$

$$C = c_1,c_2,\ldots,c_j,\ldots,c_m \tag{2}$$

To align two sequences using DTW we construct an n-by-m matrix where the (i^{th}, j^{th}) element of the matrix contains the distance $d(q_i,c_j)$ between the two points q_i and c_j (Typically the Euclidean distance is used, so $d(q_i,c_j) = (q_i - c_j)^2$). Each matrix element (i,j) corresponds to the alignment between the points q_i and c_j. This is illustrated in Figure 3. A warping path W, is a contiguous (in the sense stated below) set of matrix elements that defines a mapping between Q and C. The k^{th} element of W is defined as $w_k = (i,j)_k$ so we have:

$$W = w_1,\ w_2,\ \ldots,w_k,\ldots,w_K \qquad \max(m,n) \le K < m+n\text{-}1 \tag{3}$$

The warping path is typically subject to several constraints.

- **Boundary conditions:** $w_1 = (1,1)$ and $w_K = (m,n)$, simply stated, this requires the warping path to start and finish in diagonally opposite corner cells of the matrix.
- **Continuity:** Given $w_k = (a,b)$ then $w_{k-1} = (a',b')$ where $a–a' \le 1$ and $b-b' \le 1$. This restricts the allowable steps in the warping path to adjacent cells (including diagonally adjacent cells).
- **Monotonicity:** Given $w_k = (a,b)$ then $w_{k-1} = (a',b')$ where $a–a' \ge 0$ and $b-b' \ge 0$. This forces the points in W to be monotonically spaced in time.

There are exponentially many warping paths that satisfy the above conditions, however we are interested only in the path which minimizes the warping cost:

$$DTW(Q,C) = \min\left\{ \sqrt{\sum_{k=1}^{K} w_k} \middle/ K \right. \tag{4}$$

The K in the denominator is used to compensate for the fact that warping paths may have different lengths.

This path can be found very efficiently using dynamic programming to evaluate the following recurrence which defines the cumulative distance $\gamma(i,j)$ as the distance $d(i,j)$ found in the current cell and the minimum of the cumulative distances of the adjacent elements:

$$\gamma(i,j) = d(q_i,c_j) + \min\{\ \gamma(i\text{-}1,j\text{-}1)\ ,\ \gamma(i\text{-}1,j\ )\ ,\ \gamma(i,j\text{-}1)\ \} \tag{5}$$

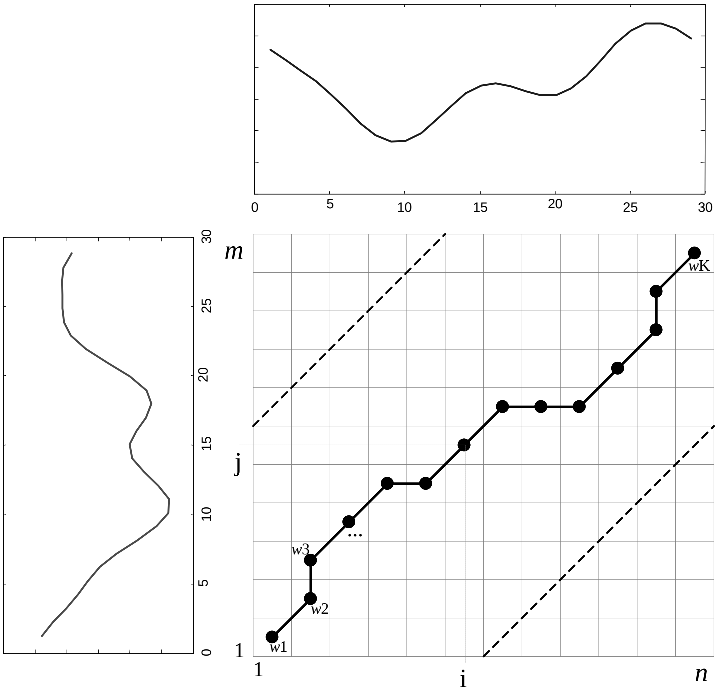

Figure 3: An example warping path.

2.1 Constraining the classic dynamic time warping algorithm

The problem of singularities was noted at least as early as 1978 (Sakoe, & Chiba 1978)). Various methods have been proposed to alleviate the problem. We briefly review them here.

1) **Windowing:** (Berndt & Clifford 1994) Allowable elements of the matrix can be restricted to those that fall into a warping window, $|i-(n/(m/j))| < R$, where R is a positive integer window width. This effectively means that the corners of the matrix are pruned from consideration, as shown by the dashed lines in Figure 3. Others have experimented with various other shaped warping windows (Rabiner et al 1978, Tappert & Das 1978, Myers et. al. 1980). This approach constrains the maximum size of a singularity, but does not prevent them from occurring.

2) **Slope Weighting:** (Kruskall & Liberman 1983,Sakoe, & Chiba 1978) If equation 5 is replaced with $\gamma(\mathrm{i,j}) = d(i,j) + \min\{ \gamma(i\text{-}1,j\text{-}1) , X\,\gamma(i\text{-}1,j) , X\,\gamma(i,j\text{-}1) \}$ where X is a positive real number, we can constrain the warping by changing the value of X. As X gets larger, the warping path is increasing biased toward the diagonal.

3) **Step Patterns (Slope constraints):** (Itakura 1975, Myers et. al. 1980) We can visualize equation 5 as a diagram of admissible step-patterns, as is Figure 4.A. The arrows illustrate the permissible steps the warping path may take at each stage.

We could replace equation 5 with $\gamma(i,j) = d(i,j) + \min\{ \gamma(i-1,j-1) , \gamma(i-1,j-2) , \gamma(i-2,j-1) \}$, which corresponds with the step-pattern show in Figure 4.B. Using this equation the warping path is forced to move one diagonal step for each step parallel to an axis. Dozens of different step-patterns have been considered, Rabiner and Juang (1993) contains a review.

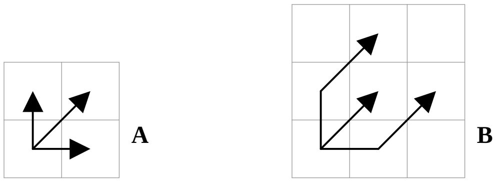

Figure 4: A pictorial representation of two alternative step-patterns:
A) The pattern corresponding to $\gamma(i,j) = d(i,j) + \min\{ \gamma(i-1,j-1) , \gamma(i-1,j) , \gamma(i,j-1) \}$
B) The pattern corresponding to $\gamma(i,j) = d(i,j) + \min\{ \gamma(i-1,j-1) , \gamma(i-1,j-2) , \gamma(i-2,j-1) \}$

All the above may help to mitigate the problem of singularities, but at the risk of missing the correct warping. Additionally, it is not obvious how to chose the various parameters (*R* for Windowing and *X* for Slope Weighting) or Step-Pattern.

3 Derivative dynamic time warping

If DTW attempts to align two sequences that are similar except for local accelerations and decelerations in the time axis, the algorithm is likely to be successful. The algorithm has problems when the two sequences also differ in the Y-axis. Global differences, affecting the entire sequences, such as different means (offset translation), different scalings (amplitude scaling) or linear trends can be efficiently removed (Keogh and Pazzani 1998, Agrawal et. al. 1995). However the two series may also have local differences in the Y-axis, for example a valley in one sequence may be deeper that the corresponding valley in the other time series. Consider Figure 5 as an example. Two identical sequences will clearly produce a one to one alignment. But if we slightly change a local feature, in this case the depth of a valley, DTW attempts to explain the difference in terms of the time-axis and produces two singularities.

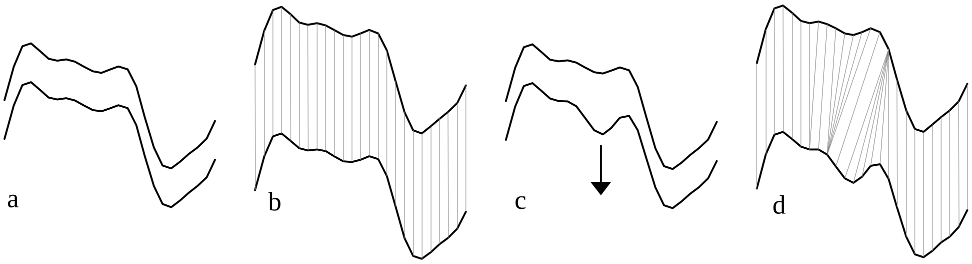

Figure 5: Using DTW, two identical sequences (a) will clearly produce a one to one alignment (b). However, if we slightly change a local feature, in this case the depth of a valley (c), DTW attempts to explain the difference in terms of the time-axis and produces two singularities (d).

The weakness of DTW is in the features it considers. It only considers a datapoints Y-axis value. For example consider two datapoints q_i and c_j which have identical values, but q_i is part of a rising trend and c_j is part of a falling trend. DTW considers a mapping between these two points ideal, although intuitively we would prefer not to map a rising trend to a falling trend. To prevent this problem we propose a modification of DTW that does not consider the Y-values of the datapoints, but rather considers the higher level feature of "shape". We obtain information about shape by considering the first derivative of the sequences, and thus call our algorithm Derivative Dynamic Time Warping (DDTW).

3.1 Algorithm details

As before we construct an n-by-m matrix where the (i^{th}, j^{th}) element of the matrix contains the distance $d(q_i,c_j)$ between the two points q_i and c_j. With DDTW the distance measure $d(q_i,c_j)$ is not Euclidean but rather the square of the difference of the estimated derivatives of q_i and c_j. While there exist sophisticated methods for estimating derivatives, particularly if one knows something about the underlying model generating the data, we use the following estimate for simplicity and generality.

$$D_x[q] = \frac{(q_i - q_{i-1}) + ((q_{i+1} - q_{i-1})/2)}{2} \qquad 1 < i < m \qquad (6)$$

This estimate is simply the average of the slope of the line through the point in question and its left neighbor, and the slope of the line through the left neighbor and the right neighbor. Empirically this estimate is more robust to outliers than any estimate considering only two datapoints. Note the estimate is not defined for the first and last elements of the sequence. Instead we use the estimates of the second and penultimate elements respectively. For noisy datasets we use exponential smoothing (Mills 1990) before attempting to estimate the derivatives.

DDTW's time complexity is $O(mn)$, which is the same as DTW. There are some added constant factors because of derivative estimating step, but there is no need to remove offset translation, a necessary step for DTW. Empirically the two algorithms take approximately the same time. A variety of optimizations have been proposed for DTW (Myers et. al 1990). We omit discussion of them here for brevity, but note that they apply equally well to DDTW.

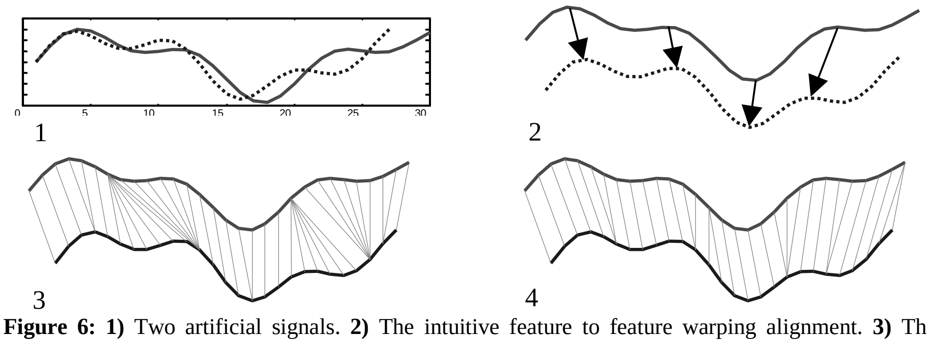

Figure 6: 1) Two artificial signals. **2)** The intuitive feature to feature warping alignment. **3)** The alignment produced by classic DTW. **4)** The alignment produced by DDTW.

4 Experimental results

We have conducted a number of experiments to compare DDTW to classic DTW. We are interested in two properties of each approach. First, does the algorithm mistakenly find warping where none exist? Secondly, can the algorithm correctly find the correct warping where it does exists?

4.1 Spurious warping

To test for spurious warping we presented both algorithms with two sequences that are similar but contain no warping of the time axis. For this purpose we needed sequences which measured related phenomena contemporaneously. Each in case, the pairs of sequences are highly correlated but not identical, in particular they contains minor (local) differences. We used the following three datasets. Samples of each are shown in Figure 7.

- **Space Shuttle:** A set of 9 sequences from sensors that measured acceleration in a particular direction during the first eight hours of Shuttle Mission STS-57. (Sensors: V71H4100B, V71H3100B, V71H3100B, V71H4140B, V71H3140B, V71H2140B, V71H4180B, V71H3180B, V71H2180B).
- **Exchange rate:** A set of 5 sequences containing the exchange rate between the Deutschmark and five other European currencies over a six month period.
- **EEG:** A set of 9 Electroencephalograph measurements that were a subset of sensors from a 21-lead reading. Each sequence contains 514 datapoints.

We normalized all sequences to have a mean of zero and a standard deviation of one. We chose these three datasets because they have widely varying properties of shape, noise, autocorrelation etc.

As noted in section 2, K, the length of the warping path is bounded such that max(m,n) ≤ K < m+n-1. All the sequences within a dataset have the same length, so we have $m = n$ and therefore $m \leq K < 2m-1$. We define W as the amount of warping implied by an algorithm:

$$W = (K - m)/m \qquad 0 \leq W < 1 \qquad (7)$$

If an algorithm discovers no warping between two sequences, W will equal zero. The more warping discovered the larger the value of W (to a maximum of $W = 1$). For each of the three datasets listed above, we compared each sequence to every other sequence and calculated the average value of W. The results are presented in Table 1.

	Mean *W* for **DTW**	Mean *W* for **DDTW**
Space Shuttle	0.17	0.03
Exchange rate	0.24	0.03
EEG	0.19	0.04

Table 1: The amount of warping implied by the two algorithms discussed in this paper.

These results confirm what a casual perusal of Figure 7 tells us. DTW attempts to correct minor differences between the sequences by wild warpings of the time axis. In contrast DDTW, which considers higher levels features is much less inclined to "find" a warping were none exists.

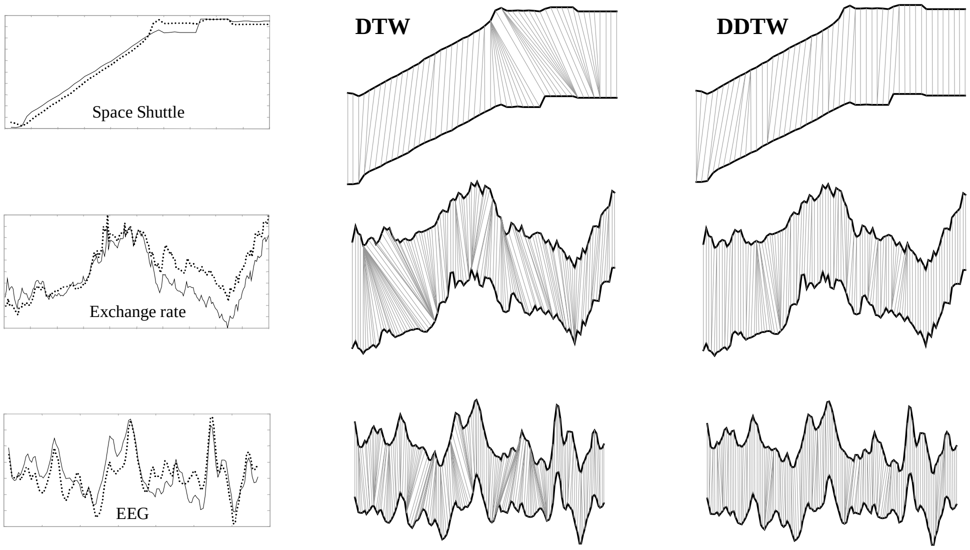

Figure 7: Examples of the three experimental datasets used in this paper. Each box in the leftmost column contains two related sequences that do *not* contain time warping, but do have minor differences in the Y-axis. The warpings "discovered" by the two algorithms discussed in this paper are therefore completely spurious. The two rightmost columns show examples of the warpings returned by both algorithms. Table 1 contains a numerical comparison.

4.2 Finding the correct warping

To test the ability of an algorithm to discover the correct warping between two sequences we need sequences for which the correct warping is known. Our approach is to take a sequence *Q* and to make a copy of it. This copy, *Q'* then has a warping randomly inserted into it. We can then use *Q* and *Q'* as input into the two algorithms and compare warpings.

To distort sequence *Q'* we begin by randomly choosing an anchor point, somewhere on the sequence. We first distort the Y-axis by adding or subtracting a Gaussian bump, centered on the anchor point, to the sequence. We did this for 3 different height bumps 0.0, 0.1 and 0.2 (0.0 corresponding to no bump). Next, we randomly shifted the anchor point 15 time-units left or right. The other datapoints were moved to compensate for this shift by an amount that depended on their inverse squared distance to the anchor point, thus localizing the effect. After the X-axis transformation we interpolated the data back onto the original, equi-spaced X-axis. The net effect of the two transformations is a smooth local distortion of the original sequence, as shown in Figure 8.

Given the above, the correct alignment between any two points in the two sequences Q and Q' is known, we denote this as a double arrow $q_i \Leftrightarrow q'_j$. The actual alignment returned by an algorithm we denote as a single arrow $q_i \leftrightarrow q'_{j'}$. We can summarize the amount of misalignment M, by summing the differences between the correct alignment and the returned alignment.

$$M = \frac{\sum_{i=1}^{m} |j - j'|}{\frac{1}{2} m(m-1)} \qquad where \quad q_i \leftrightarrow q'_{j'},\ and\ q_i \Leftrightarrow q'_j \qquad 0 \le M \le 1 \ (8)$$

The denominator in equation (8) is simply to normalize for different length sequences. For each of the three datasets listed above, we took each sequence and randomly inserted a warping, then measured the amount of misalignment. We repeated this ten times for every sequence at each bump height and averaged the results, which are recorded in Table 2. Because there was little difference between the three datasets, we pooled their results.

Bump Height	Mean *M* for **DTW**	Mean *M* for **DDTW**
0.0 (no bump)	0.0043	0.0034
0.1	0.0547	0.0039
0.2	0.1278	0.0053

Table 2: A comparison of the misalignment created by the two algorithms discussed in this paper.

The results show that while there is little difference between the algorithms when the distortion is confined to the X-axis, even modest amounts of distortion to the Y-axis cause DTW to degrade rapidly.

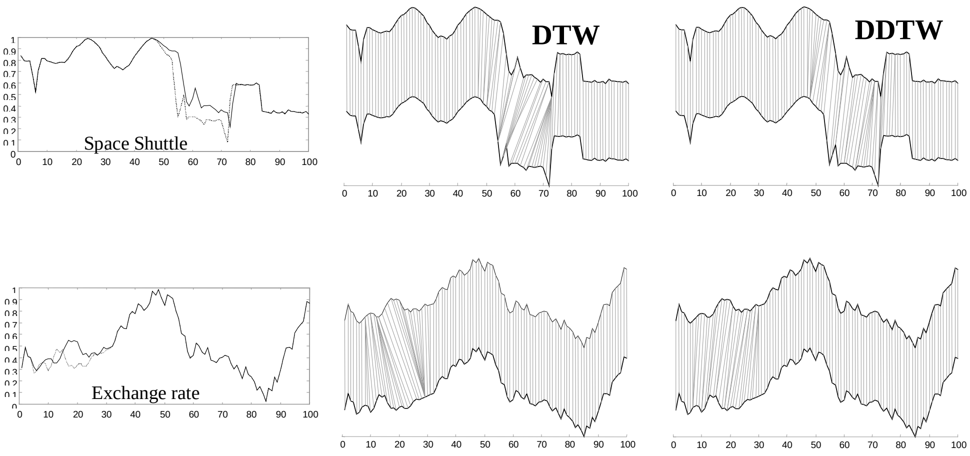

Figure 8: Each box in the leftmost column contains a sequence and an artificially distorted version of it. The two rightmost columns show examples of the warpings returned by both algorithms. Table 2 contains a numerical comparison.

5 Conclusions and future work

We have described a modification of dynamic time warping and shown it produces superior alignments between time series. In the future we hope to extend the technique to higher level representations of time series such as piecewise linear segments (Keogh and Pazzani 1998) or Fourier transforms (Agrawal et. al. 1995), thereby mitigating the algorithms time and space complexity.

Acknowledgments

The authors would like to acknowledge Mohammed Waleed Kadous, Matt Schmill and Tim Oates for their useful comments on the early stages of this work.

References

Agrawal, R., Lin, K. I., Sawhney, H. S., & Shim, K. (1995). Fast similarity search in the presence of noise, scaling, and translation in times-series databases. In *VLDB*, September.

Berndt, D. & Clifford, J. (1994) Using dynamic time warping to find patterns in time series. *AAAI-94 Workshop on Knowledge Discovery in Databases (KDD-94),* Seattle, Washington.

Caiani, E.G., Porta, A., Baselli, G., Turiel, M., Muzzupappa, S., Pieruzzi, F., Crema, C., Malliani, A. & Cerutti, S. (1998) Warped-average template technique to track on a cycle-by-cycle basis the cardiac filling phases on left ventricular volume. IEEE Computers in Cardiology. Vol. 25 Cat. No.98CH36292, NY, USA.

Gavrila, D. M. & Davis,L. S.(1995). Towards 3-d model-based tracking and recognition of human movement: a multi-view approach*. In International Workshop on Automatic Face- and Gesture-Recognition*. IEEE Computer Society, Zurich.

Gollmer, K., & Posten, C. (1995) Detection of distorted pattern using dynamic time warping algorithm and application for supervision of bioprocesses. On-Line Fault Detection and Supervision in the Chemical Process Industries (Edited by: Morris, A.J.; Martin, E.B.).

Itakura, F. (1975). Minimum prediction residual principle applied to speech recognition. *IEEE Trans. Acoustics, Speech, and Signal Proc.*, Vol. ASSP-23, 52-72.

Keogh, E., & Pazzani, M. (1998). An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. *Proceedings of the 4rd International Conference of Knowledge Discovery and Data Mining*. pp 239-241, AAAI Press.

Keogh, E., & Pazzani, M. (2000) Scaling up dynamic time warping for datamining applications*. In 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Boston.

Kruskall, J. B. & Liberman, M. (1983). The symmetric time warping algorithm: From continuous to discrete. In *Time Warps, String Edits and Macromolecules: The Theory and Practice of String Comparison*. Addison-Wesley.

Mills, T., C. (1990). Time Series Techniques for Economists, Cambridge University Press.

Myers, C., Rabiner, L & Roseneberg, A. (1980). Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. *IEEE Trans. Acoustics, Speech, and Signal Proc*., Vol. ASSP-28, 623-635

Rabiner, L. & Juang, B. (1993). Fundamentals of speech recognition. Englewood Cliffs, N.J, Prentice Hall.

Rabiner, L., Rosenberg, A. & Levinson, S. (1978). Considerations in dynamic time warping algorithms for discrete word recognition. *IEEE Trans. Acoustics, Speech, and Signal Proc*., Vol. ASSP-26, 575-582.

Sakoe, H. & Chiba, S. (1978) Dynamic programming algorithm optimization for spoken word recognition*. IEEE Trans. Acoustics, Speech, and Signal Proc*., Vol. ASSP-26, 43-49.

Schmill, M., Oates, T. & Cohen, P. (1999). Learned models for continuous planning. In *Seventh International Workshop on Artificial Intelligence and Statistics*.

Strik, H. & Boves, L. (1988) Averaging physiological signals with the use of a DTW algorithm. *In Proceedings SPEECH'88, 7th FASE Symposium*, Edinburgh, Book 3, 883-890.

Tappert, C. & Das, S. (1978). Memory and time improvements in a dynamic programming algorithm for matching speech patterns. *IEEE Trans. Acoustics, Speech, and Signal Proc*., Vol. ASSP-26, 583-586.

Yi, B., Jagadish, H and Faloutsos, C. (1998) Efficient retrieval of similar time sequences under time warping. *In International Conference of Data Engineering*.