Int. J. Appl. Math. Comput. Sci., *2019, Vol. 29, No. 2, 375–392*
DOI: 10.2478/amcs-2019-0028

TIMES SERIES AVERAGING AND DENOISING FROM A PROBABILISTIC PERSPECTIVE ON TIME–ELASTIC KERNELS

PIERRE-FRANCOIS MARTEAU [a]

[a] Institute for Research in Computer Science and Stochastic Systems (IRISA)
University of Southern Brittany, Tohannic Campus, 56000 Vannes, France
e-mail: pierre-francois.marteau@univ-ubs.fr

In the light of regularized dynamic time warping kernels, this paper re-considers the concept of a time elastic centroid for a set of time series. We derive a new algorithm based on a probabilistic interpretation of kernel alignment matrices. This algorithm expresses the averaging process in terms of stochastic alignment *automata*. It uses an iterative agglomerative heuristic method for averaging the aligned samples, while also averaging the times of their occurrence. By comparing classification accuracies for 45 heterogeneous time series data sets obtained by first nearest centroid/medoid classifiers, we show that (i) centroid-based approaches significantly outperform medoid-based ones, (ii) for the data sets considered, our algorithm, which combines averaging in the sample space and along the time axes, emerges as the most significantly robust model for time-elastic averaging with a promising noise reduction capability. We also demonstrate its benefit in an isolated gesture recognition experiment and its ability to significantly reduce the size of training instance sets. Finally, we highlight its denoising capability using demonstrative synthetic data. Specifically, we show that it is possible to retrieve, from few noisy instances, a signal whose components are scattered in a wide spectral band.

Keywords: time series averaging, time elastic kernel, dynamic time warping, hidden Markov model, classification, denoising.

1. Introduction

Since Maurice Fréchet's pioneering work (Fréchet, 1906) in the early 1900s, *time-elastic* matching of time series or symbolic sequences has attracted much attention from the scientific community in numerous fields such as information indexing and retrieval, pattern analysis, extraction and recognition, data mining, etc. This approach has impacted a very wide spectrum of applications addressing socio-economic issues such as the environment, industry, health, energy, defense and so on.

Among other time elastic measures, dynamic time warping (DTW) was widely popularized during the 1970s with the advent of speech recognition systems (Velichko and Zagoruyko, 1970; Sakoe and Chiba, 1971), along with numerous variants that have since been proposed to match time series with a certain degree of time distortion tolerance.

The main issue addressed in this paper is time series or shape averaging in the context of a time elastic distance. Time series averaging or signal averaging is a long-standing issue that is currently becoming increasingly prevalent in the big data context; it is relevant for denoising (Kaiser and Knight, 1979; Hassan and Anwar, 2010), summarizing subsets of time series (Petitjean *et al.*, 2011), defining significant prototypes, identifying outliers (Gupta *et al.*, 2014), performing data mining tasks (mainly exploratory data analysis such as clustering) and speeding up classification (Petitjean *et al.*, 2014), as well as regression or data analysis processes in a big data context.

In this paper, we specifically tackle the question of averaging subsets of time series, not from considering the DTW measure itself, as has already been largely exploited, but from the perspective of the so-called regularized DTW kernel (KRDTW). From this new viewpoint, the estimation of a time series average or centroid can be readily addressed with a probabilistic interpretation of kernel alignment matrices allowing a precise definition of the average of a pair of time series from the expected value of local alignments of samples. The tests carried out so far demonstrate the robustness and efficiency of this approach compared with the state-of-the-art one.

The structure of this paper is as follows. Following the introductory section, Section 2 summarizes the most relevant related studies on time series averaging as well as DTW kernelization. In Section 3, we derive a probabilistic interpretation of kernel alignment matrices evaluated on a pair of time series by establishing a parallel with a forward-backward procedure on a stochastic alignment automaton. In the fourth section, we define the average of a pair of time series based on the alignment expectation of pairs of samples, and we propose an algorithm designed for the averaging of any subset of time series using a pairwise aggregating procedure. We present in the Section 5 three complementary experiments to assess our approach against the state of the art, and then conclude the paper.

2. Related works

Time series averaging in the context of (multiple) time elastic distance alignments has been mainly addressed in the scope of the dynamic time warping (DTW) measure (Velichko and Zagoruyko, 1970; Sakoe and Chiba, 1971). Although other time elastic distance measures such as the edit distance with real penalty (ERP) (Chen and Ng, 2004) or the time warp edit distance (TWED) (Marteau, 2009) could be considered instead, without loss of generality, we remain focused throughout this paper on DTW and its kernelization.

2.1. DTW and a time elastic average of a pair of time series. A classical formulation of DTW can be given as follows. If d is a fixed positive integer, we define a time series of length n as a multidimensional sequence $o_1^n = o_1 o_2 \dots o_n$, such that $\forall i \in \{1, \dots, n\}$, $o_i \in \mathbb{R}^d$.

Definition 1. If o_1^n and ${o'}_1^{n'}$ are two time series with respective lengths n and n', an *alignment path* $\pi = (\pi_k)$ of length $p = |\pi|$ between o_1^n and ${o'}_1^{n'}$ is represented by a sequence

$$\pi : \{1, \dots, p\} \rightarrow \{1, \dots, n\} \times \{1, \dots, n'\}$$

such that $\pi_1 = (1, 1)$, $\pi_p = (n, n')$, and using the notation $\pi_k = (i_k, j_k)$, for all $k \in \{1, \dots, p-1\}$, $\pi_{k+1} = (i_{k+1}, j_{k+1}) \in \{(i_k + 1, j_k), (i_k, j_k + 1), (i_k + 1, j_k + 1)\}$.

If δ is a distance on $\mathbb{R}^d$, the global *cost* of a warping path π is the sum of distances (or squared distances or local costs) between pairwise elements of the two time series along π, i.e.,

$$\mathrm{cost}(\pi) = \sum_{(i_k, j_k) \in \pi} \delta(o_{i_k}, {o'}_{j_k}).$$

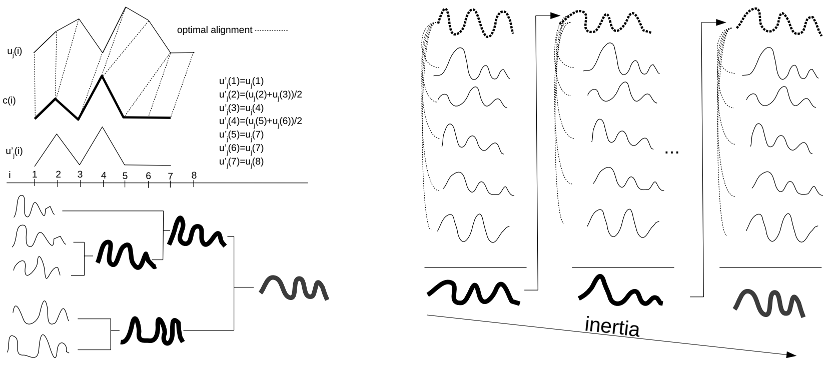
Definition 2. For a pair of finite time series o_1^n and ${o'}_1^{n'}$, any warping path has a finite length, and thus the number of existing warping paths is finite. Hence, there exists at least one path π^* whose cost is minimal, so we can define $\mathrm{DTW}(o, o')$ as the minimal cost taken over all existing warping paths. Hence

$$\begin{aligned} \mathrm{DTW}(o_1^n, {o'}_1^{n'}) &= \min_{\pi} \mathrm{cost}(\pi(o_1^n, {o'}_1^{n'})) \\ &= \mathrm{cost}(\pi^*(o_1^n, {o'}_1^{n'})). \end{aligned} \tag{1}$$

Definition 3. From the DTW measure, Gupta *et al.* (1996) have defined the time elastic average $a(o, o')$ of a pair of time series o_1^n and ${o'}_1^{n'}$ as the time series $A_1^{|\pi^*|}$ whose elements are $A_k = \mathrm{mean}(o_{\pi_k^*(1)}, {o'}_{\pi_k^*(2)})$, $\forall k \in 1, \dots, |\pi^*|$, where *mean* corresponds to the definition of the mean in the Euclidean space.

2.2. Time elastic centroid of a set of time series. A single alignment path is required to calculate the time elastic centroid of a pair of time series (Definition 1). However, multiple path alignments need to be considered to evaluate the centroid of a larger set of time series. Multiple alignments have been widely studied in bioinformatics (Fasman and Salzberg, 1998), and it has been shown that determining the optimal alignment of a set of sequences under the score scheme of the sum of all pairs (SP) is an NP-complete problem (Wang and Jiang, 1994; Just and Just, 1999). The time and space complexity of this problem is $O(L^k)$, where k is the number of sequences in the set and L is the length of the sequences when using dynamic programming to search for an optimal solution (Carrillo and Lipman, 1988). This result applies to the estimation of the time elastic centroid of a set of k time series with respect to the DTW measure. Since the search for an optimal solution rapidly becomes intractable with increasing k, sub-optimal heuristic solutions have been subsequently proposed, most of them falling into one of the following three categories.

2.2.1. Progressive heuristics. Progressive heuristic methods estimate the time elastic centroid of a set of k time series by combining pairwise centroids (Definition 3). This kind of approach constructs a binary tree whose leaves correspond to the time series of the data set and nodes to the calculation of a local pairwise centroid such that, when the tree is complete, the root is associated with the estimated data set centroid. The proposed strategies differ in the way the tree is constructed. One popular approach consists in providing a random order for the leaves, and then constructing the binary tree up to the root using this ordering (Gupta *et al.*, 1996). Another approach involves constructing a dendrogram (a hierarchical ascendant clustering) from



(a) (b)

Fig. 1. Pairwise averaging ((a) top) and progressive hierarchical agglomeration ((a) bottom) vs. iterative agglomeration strategies (b). Final centroid approximations are presented as thick lines. Temporary estimates are presented using a thick dotted black line.

the data set and then using this dendrogram to calculate pairwise centroids starting with the closest pairs of time series and progressively aggregating series that are farther away (Niennattrakul and Ratanamahatana, 2009), as illustrated on the left of Fig. 1(a). Note that these heuristic methods are entirely based on the calculation of a pairwise centroid, so they do not explicitly require the evaluation of a DTW centroid for more than two time series. Their degree of complexity varies linearly with the number of time series in the data set.

2.2.2. Iterative heuristics. Iterative heuristics are based on an iterated three-step process. For a given temporary centroid candidate, the first step consists of calculating the inertia, i.e., the sum of the DTW distances between the temporary centroid and each time series in the data set. The second step (Fig. 1(b)) evaluates the best pairwise alignment with the temporary centroid c_1^L of length L for each time series ${}^jo_1^{n_j}$ in the data set ($j \in \{1, \dots, N\}$). A new time series ${}^jo'^{n_j}_1$ is thus constructed that contains the contributions of all the samples of time series ${}^jo_1^{n_j}$, but with time being possibly stretched (duplicate samples) or compressed (average of successive samples) according to the best alignment path as exemplified in Fig. 1(b). The third step consists in producing a new temporary centroid candidate c'^L_1 from the set $\{{}^jo'^{n_j}_1\}$ by successively averaging (in the sense of the Euclidean centroid) the samples at every timestamp i of the time series ${}^jo'^{n_j}_1$. Basically, we have $c'_i = (1/n)\sum_{j=1,\dots,n} {}^jo'_i$.

Then, the new centroid candidate replaces the previous one and the process is iterated until the inertia is no longer reduced or the maximum number of iterations is reached. Generally, the first temporary centroid candidate is taken as the DTW medoid of the data set considered. This process is illustrated on the left of Fig. 1. The three steps of this heuristic method were first proposed by Abdulla *et al.* (2003). The iterative aspect of this heuristic approach was initially introduced by Hautamaki *et al.* (2008) and refined by Petitjean *et al.* (2011), who introduced the DTW barycenter averaging (DBA) algorithm. Note that, in contrast to the progressive method, this kind of approach needs to evaluate, at each iteration, all the alignments with the current centroid candidate. The complexity of the iterative approach is higher than the progressive one, the extra computational cost being linear with the number of iterations. More sophisticated approaches have been proposed to escape from some local *minima*. For instance, Petitjean and Gançarski (2012) evaluated a genetic algorithm for managing a population of centroid candidates, thus improving, with some success, the straightforward iterative heuristic methods.

2.2.3. Optimization approaches. Given the entire set of time series $\mathbb{S}$ and a subset of n time series $S = \{{}^jo_1^{n_j}\}_{j=1,\dots,n} \subseteq \mathbb{S}$, optimization approaches attempt to estimate the centroid of S from the definition of an optimization problem, which is generally expressed by

$$c_1^{n_c} = \arg\min_{s_1^{n_s} \in S} \sum_{j=1}^{n} \mathrm{DTW}(s_1^{n_s}, {}^jo_1^{n_j}). \tag{2}$$

Among other works, some attempts to use this kind of direct approach for the estimation of time elastic centroid were recently made by Zhou and De la Torre (2009; 2016) and Soheily-Khah *et al.* (2016).

Zhou and De la Torre (2009) detail a canonical time warp (CTW) and a generalized version of it (GCTW) (Zhou and De la Torre, 2016) that combines DTW and CCA (canonical correlation analysis) for temporally aligning multi-modal motion sequences. From a least-squares formulation for DTW, a non-convex optimization problem is handled by means of a coordinate-descent approach that alternates between multiple temporal alignments using DTW (or a variant exploiting a set of basis functions to parameterize the warping paths) and spatial projections using CCA (or a multi-set extension of CCA). Whilst these approaches have not been designed to explicitly propose a centroid estimation, they do provide multi-alignment paths that can straightforwardly be used to compute a centroid estimate. As an extension to CTW, GCTW requires the set-up of generally "smooth" function basis that constrains the shape of the admissible alignment paths. This ensures the computational efficiency of GCTW, but in return it may induce some drawback, especially when considering the averaging of "unsmoothed" time series that may involve very "jerky" alignment paths. The choice of this function basis may require some expertise in the data.

Soheily-Khah *et al.* (2016) derived a non-convex constrained optimization problem by integrating a temporal weighting of local sample alignments to highlight the temporal region of interest in a time series data set, thus penalizing the other temporal regions. Although the number of parameters to optimize is linear with the size and the dimensionality of the time series, the two-step gradient-based optimization process they derived is very computationally efficient and shown to outperform the state of the art approaches on some challenging scalar and multivariate data sets. However, as numerous local *optima* exist in practice, the method is not guaranteed to converge towards the best possible centroid, which is anyway the case in all other approaches. Furthermore, their approach, due to combinatorial explosion, cannot be adapted for time elastic kernels like the one addressed in this paper and described in Section 2.4.

2.3. Discussion and motivation. According to the state of the art in time elastic centroid estimation, an exact centroid, if it exists, can be calculated by solving an NP-complete problem whose complexity is exponential with the number of time series to be averaged. Heuristic methods with increasing time complexity have been proposed since the early 2000s. Simple pairwise progressive aggregation is a less complex approach, but it suffers from dependence on initial conditions. Iterative aggregation is reputed to be more efficient, but it entails a higher computational cost. It could be combined with ensemble methods or soft optimization such as genetic algorithms. The non-convex optimization approach has the merit of directly addressing the mathematical formulation of the centroid problem in a time elastic distance context. This approach nevertheless involves a higher complexity and must deal with a relatively large set of parameters to be optimized (the weights and the sample of the centroid). Its scalability could be questioned, specifically for high dimensional multivariate time series.

It should also be mentioned that some criticism of these heuristic methods was made by Niennattrakul and Ratanamahatana (2007). Among other drawbacks, the fact that DTW is not a metric could explain the occurrence of unwanted behaviors such as a centroid drift outside the time series cluster to be averaged. We should also bear in mind that keeping a single best alignment can increase the dependence of the solution on the initial conditions. It may also increase the aggregating order of the time series proposed by the chosen method, or potentially enhance the convergence rate.

In this study, we do not directly address the issue of time elastic centroid estimation from the DTW perspective, but rather from the point of view of the regularized dynamic time warping kernel (KRDTW) (Marteau and Gibet, 2014). Although this perspective allows us to consider centroid estimation as a preimage problem, which is in itself another optimization perspective, we rather show that computation of KRDTW alignment matrices can be described as the result of applying a forward-backward algorithm on stochastic alignment automata. This probabilistic interpretation of the pairwise alignment of time series makes us propose a robust averaging scheme for any set of time series that interpolate jointly along the time axis and in the sample space. Furthermore, this scheme significantly outperforms the current state of the art method, as shown by our experiments.

2.4. Time elastic kernels and their regularization. The dynamic time warping (DTW) distance between two time series $o_1^p = o_1 o_2 \cdots o_p$ and $o'^q_1 = o'_1 o'_2 \cdots o'_q$ of lengths p and q, respectively (Velichko and Zagoruyko, 1970; Sakoe and Chiba, 1971), as defined in Eqn. (1), can be recursively evaluated as

$$d_{dtw}(o_1^p, o'^q_1) = d_E^2(o_p, o'_q) + \min \begin{cases} d_{dtw}(o_1^{p-1}, o'^q_1) \\ d_{dtw}(o_1^{p-1}, o'^{q-1}_1) \\ d_{dtw}(o_1^p, o'^{q-1}_1), \end{cases} \tag{3}$$

where $d_E(o_p, o'_q)$ is the Euclidean distance defined on $\mathbb{R}^d$ between the two positions in sequences o_1^p and o'^q_1 taken at times p and q, respectively.

Apart from the fact that the triangular inequality does not hold for the DTW distance measure, it is not possible to define a positive definite kernel directly from this distance. Hence, the optimization problem, which is inherent to the learning of a kernel machine, is no longer convex and could be a source of limitation due to the emergence of local minima.

Regularized DTW. The seminal work by Cuturi *et al.* (2007), prolonged recently by Marteau and Gibet (2014), leads us to propose new guidelines to ensure that kernels constructed from elastic measures such as DTW are positive definite. A simple instance of such a regularized kernel, derived from the work of Marteau and Gibet (2014), can be expressed as a convolution kernel, which makes use of two recursive terms:

$$
\begin{aligned}
&\mathrm{KRDTW}(o_1^p, {o'}_1^q) = K_{rdtw}(o_1^p, {o'}_1^q) + K'_{rdtw}(o_1^p, {o'}_1^q),\\
&K_{rdtw}(o_1^p, {o'}_1^q)\\
&\quad = \frac{1}{3} e^{-\nu d_E^2(o_p, {o'}_q)} \Big(h(p-1,q) K_{rdtw}(o_1^{p-1}, {o'}_1^q)\\
&\qquad + h(p-1,q-1) K_{rdtw}(o_1^{p-1}, {o'}_1^{q-1})\\
&\qquad + h(p,q-1) K_{rdtw}(o_1^{p}, {o'}_1^{q-1}) \Big),\\
&\\
&K'_{rdtw}(o_1^p, {o'}_1^q)\\
&\quad = \frac{1}{3} \Big(h(p-1,q) K'_{rdtw}(o_1^{p-1}, {o'}_1^q) e^{-\nu d_E^2(o_p, {o'}_p)}\\
&\qquad + \frac{1}{2} h(p-1,q-1) K'_{rdtw}(o_1^{p-1}, {o'}_1^{q-1})\\
&\qquad \times \left(e^{-\nu d_E^2(o_p, {o'}_p)} + e^{-\nu d_E^2(o_q, {o'}_q)} \right)\\
&\qquad + h(p,q-1) K'_{rdtw}(o_1^{p}, {o'}_1^{q-1}) e^{-\nu d_E^2(o_q, {o'}_q)} \Big),
\end{aligned}
\tag{4}
$$

where $\nu \in \mathbb{R}^+$ is a *stiffness* parameter which weighs the local contributions, i.e., the distances between locally aligned positions, $d_E(\cdot,\cdot)$ is a distance defined on $\mathbb{R}^d$, and h is a symmetric binary non-negative function, usually with values in $\{0,1\}$, used, e.g., to define a symmetric corridor around the main diagonal to limit the "time elasticity" of the kernel. For the remainder of the paper we will not consider any corridor; hence $h(\cdot,\cdot) = 1$ everywhere.

The initialization is simply $K_{rdtw}(o_1^0, {o'}_1^0) = K'_{rdtw}(o_1^0, {o'}_1^0) = 1$.

The first term, K_{rdtw}, is very close to the global alignment kernel, K_{ga}, proposed by Cuturi *et al.* (2007). The second term, ${K'}_{rdtw}$, is a regularization term that allows defining a positive definite kernel k_π for each of the admissible alignment paths π. Hence, the global KRDTW kernel can be seen as the sum on a set of the admissible alignment paths (or any subset of it) of the k_π kernels.

The main idea behind this regularization is to replace the operators $\min$ and $\max$ (which prevent symmetrization of the kernel) by a summation operator. This allows us to consider the best possible alignment, as well as all the best (or nearly the best) paths by summing their overall cost. The parameter ν is used to check what is termed a nearly-the-best alignment, thus penalizing alignments that are too far away from the optimal ones. This parameter can be easily optimized through a cross-validation.

For each alignment path, KRDTW evaluates the product of local alignment costs $e^{-\nu d_E^2(o_p, {o'}_q))} \leq 1$ occurring along the path. This product can be very small depending on the size of the time series and the selected value for ν. This is the source for a diagonal dominance problem in the Gram matrix. But, above all, this requires to balance the choice of the ν value according to the lengths of the matched time series. This is the main (and probably the only) limitation of the KRDTW kernel: the selectivity or bandwidth of the local alignment kernels needs to be adjusted according to the lengths of the matched time series.

3. Stochastic alignment process

To introduce a probabilistic paradigm to the time elastic averaging of time series, we first consider the pairwise alignment process as the output of stochastic automata. The stochastic alignment process that we propose finds its roots in the forward-backward algorithm defined for the learning of hidden Markov models (HMMs) (Rabiner, 1989) and in the parallel between HMMs and DTW that is proposed by Juang (1985) or Nakagawa and Nakanishi (1989), and in a more distant way by Chudova *et al.* (2003). However, we differ from these founding works (and others) in the following:

1. We do not construct a parallel with DTW, but with its kernelized variant KRDTW.

2. Nakagawa and Nakanishi (1989) only consider an optimal alignment path (exploiting the Viterbi algorithm) while we consider the whole set of possible alignments (as Juang (1985)).

3. Juang (1985) constructs an asymmetric classical left-right HMM (one time series of the observation sequence, while the other plays the role of the state sequence). With a similar idea, Chudova *et al.* (2003) propose a generative mixture model along a discrete time grid axis with a local and global time warp capability. We construct instead an alignment process that conforms to the DTW recursive definition without any other hypothesis on the structure of the automata, and for which the two aligned time series play the role of the observation sequence while the set of states corresponds to that of all possible sample pair alignments.

3.1. Pairwise alignment of time series as a Markov model.

Let $o_1^n = o_1 o_2 \cdots o_n$ and ${o'}_1^{n'} = o'_1 o'_2 \cdots o'_{n'}$ be two discrete time series (observations) of lengths n and n', respectively. We suppose that these series end with a 'null' (ϵ) symbol respectively at indexes $n+1$ and $n'+1$. To align these two time series, we define a stochastic alignment automaton as a hidden Markov model. First we consider the set of states $\mathcal{S} = \{S_{1,1}, S_{1,2}, \ldots, S_{n,n'}, S_{n+1,n'+1}\}$. Each $S_{i,j}$

characterizes the alignment between the observed samples o_i and o'_j. $S_{n+1,n'+1}$ is the final (or output) state.

The hidden state variable (at 'step' τ) z_τ takes values from the set of states $\mathcal{S}$. The step index τ follows an admissible alignment path (cf. Eqn. (1)) and will take values in $\{(1,1),\ldots,(n,n')\}$. In the following, we adopt the convention that if $\tau=(i,j)$, $\tau+1\in\{(i,j+1),(i+1,j),(i+1,j+1)\}$ and $\tau-1\in\{(i,j-1),(i-1,j),(i-1,j-1)\}$.

The posterior probability for the process to be at state $S_{i,j}$ at step τ, given the sequences of observations o_1^n and ${o'}_1^{n'}$, is $P(z_\tau=S_{i,j}|o_1^n,{o'}_1^{n'})$.

The transition probabilities (which are stationary, hence independent of τ) between states are driven by a tensor $\mathbf{A}=[a_{ij;kl}]$, where, for all τ, $a_{ij;kl}=P(z_{\tau+1}=S_{k,l}|z_\tau=S_{i,j})$, $\forall(k,l)$ and $(i,j)\in\{1\cdots n\}\times\{1\cdots n'\}$. $\mathbf{A}$ can be defined according to the standard DTW definition, namely,

$$a_{ij;kl} = \begin{cases} \frac{1}{3} & \text{if } \begin{cases} (k=i \text{ and } l=j+1) \\ \text{or } (k=i+1 \text{ and } l=j+1) \\ \text{or } (k=i+1 \text{ and } l=j), \end{cases} \\ 1 & \text{if } i=k=n+1 \text{ and } j=l=n'+1, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The factor $1/3$ ensures that the transition matrix equivalent to $\mathbf{A}$ is stochastic, basically,

$$\forall i,j \sum_{kl} a_{ij;kl}=1. \tag{6}$$

For the same reason, the final state transition $S_{n+1,n'+1}$ needs to be 'looped' with $P(z_\tau=S_{n+1,n'+1}|z_{\tau-1}=S_{n+1,n'+1})=1$.

Notice that any tensor $\mathbf{A}$ satisfying (6) could be considered at this level instead of the previous DTW surrogate tensor.

Furthermore, each state is observable through the so-called emission probabilities, which are defined by a set of functions $b_{ij}(x,y)$, where (x,y) is the observation. The emission probabilities depend only on the state, not τ, and we can write $b_{ij}(x,y)=P(x,y|z_\tau=S_{i,j})=P(x,y|S_{i,j}),\forall(x,y)\in\mathbb{R}^d\times\mathbb{R}^d$ and $(i,j)\in\{1,\ldots,n\}\times\{1,\ldots,n'\}$. The b_{ij} functions are normalized such that $\iint_{x,y} b_{ij}(x,y)\,\mathrm{d}x\,\mathrm{d}y=1$. Finally, we impose that $b_{n+1,n'+1}=1$ such that the end 'loop' has no effect on the marginalized probabilities.

Here we differ from the classical HMM: the first difference lies in the nature of the observation sequence itself. Unlike the HMM, our observation consists of a pair of subsequences that are not traveled necessarily synchronously, but according to the structure of the transition tensor $\mathbf{A}$. For instance, given the DTW tensor described by (5), from the current state associated to the alignment (o_u,o'_v), three possible alignments can be reached at the next transition: (o_{u+1},o'_v), (o_u,o'_{v+1}) or (o_{u+1},o'_{v+1}).

The second difference is that the step index τ determines the state of the process, i.e., $z_\tau=S_{u,v}$ means that $\tau=(u,v)$.

The third difference from the classical HMM is that the emission probabilities are independent of the state, such that $\forall i,j\ b_{i,j}(x,y)=b(x,y)$. We use a local (density) kernel to estimate these probabilities as follows:

$$b(x,y)=\kappa(x,y)=\gamma e^{-\nu d_E^2(x,y)}, \tag{7}$$

where γ is the normalization coefficient.

Consequently, given two observation sequences o_1^n and ${o'}_1^{n'}$, we define the emission probability matrix $\mathbf{B}=[b_{kl}]=b(o_k,o'_l)=\gamma e^{-\nu d_E^2(o_k,o'_l)}$, for $k\in\{1,\ldots,n\}$ and $l\in\{1,\ldots,n'\}$.

Finally, let $\mathbf{u}$ be the initial probability vector defined by $\forall(i,j)\in\{1,\ldots,n\}\times\{1,\ldots,n'\}$, $\mathbf{u}_{ij}=P(z_1=S_{i,j})$. $\mathbf{u}_{ij}=1$ if $i=j=1$, and 0 otherwise.

Thereby, the stochastic alignment automaton is fully specified by the triplet $\theta=(\mathbf{A},\mathbf{B},\mathbf{u})$, where $\mathbf{A}$ only depends on the lengths n and n' of the observations, and $\mathbf{B}$ depends on the complete pair of observations o_1^n and ${o'}_1^{n'}$.

3.2. Forward-backward alignment algorithm.

We derive the forward-backward alignment algorithm for our stochastic alignment automaton from its classical derivation that was defined for hidden Markov models (Rabiner, 1989).

For all $S_{t,t'}\in\mathcal{S}$, the posterior probability $P(z_\tau=S|o_1^n,{o'}_1^{n'},\theta)$ is decomposed into forward/backward recursions as follows:

$$\begin{aligned} &P(z_\tau=S_{t,t'}|o_1^n,{o'}_1^{n'},\theta) \\ &\quad=\frac{P(o_1^n,{o'}_1^{n'},z_\tau=S_{t,t'}|\theta)}{P(o_1^n,{o'}_1^{n'}|\theta)} \\ &\quad=\frac{P(o_1^t,o_t^n,{o'}_1^{t'},{o'}_{t'}^{n'},z_\tau=S_{t,t'}|\theta)}{P(o_1^n,{o'}_1^{n'}|\theta)} \\ &\quad=\frac{P(o_t^n,{o'}_{t'}^{n'}|z_\tau=S_{t,t'},\theta)P(z_\tau=S_{t,t'},o_1^t,{o'}_1^{t'}|\theta)}{P(o_1^n,{o'}_1^{n'}|\theta)}. \end{aligned} \tag{8}$$

The last equality results from the application of the Bayes rule and the conditional independence of $o_t^n,{o'}_{t'}^{n'}$

and $o_1^t, {o'}_1^{t'}$ given S, θ.

Let $\alpha_\tau(t,t') = P(o_1^t, {o'}_1^{t'}, z_\tau = S_{t,t'}|\theta)$ be the probability for the process to be at state $S_{t,t'}$ at step τ once the partial observation sequences $(o_1^t, {o'}_1^{t'})$ are aligned. Here $\alpha_\tau(t,t')$ can be recursively evaluated as the forward procedure

$$\begin{cases} \alpha_{\tau_I}(1,1) = u_{11}b_{11}, \\ \alpha_\tau(t,t') = b_{tt'} \sum\limits_{\tau-1 \in \mathcal{F}_{t,t'}} \alpha_{\tau-1}(u,v) a_{uv;tt'}, \end{cases} \tag{9}$$

where $\tau_I = (1,1)$, $\mathcal{F}_{t,t'}$ is the subset of indexes of states allowing reaching the state $S_{t,t'}$ in a single transition. For the DTW tensor $\mathbf{A}$ (cf. (5)), we have $\mathcal{F}_{t,t'} = \{(t-1,t'), (t,t'-1), (t-1,t'-1)\}$, whenever $t < n+1$ and $t' < n'+1$.

Similarly, let $\beta_\tau(t,t') = P(o_t^n, {o'}_{t'}^{n'}|z_\tau = S_{t,t'}, \theta)$ be the probability of the alignment of the pair of partial sequences $(o_t^n, {o'}_{t'}^{n'})$ given the alignment process is at state $S_{t,t'}$ at step τ. Here $\beta_\tau(t,t')$ can be recursively evaluated as the backward procedure

$$\begin{cases} \beta_{\tau_F}(n,n') = 1, \\ \beta_\tau(t,t') = \sum\limits_{\tau+1 \in \mathcal{B}_{t,t'}} \beta_{\tau+1}(u,v) a_{tt';uv} b_{uv}, \end{cases} \tag{10}$$

where $\tau_F = (n,n')$, $\mathcal{B}_{t,t'}$ is the subset of indexes of states that can be reached from the state $S_{t,t'}$ in a single transition, and $T = n + n' + 2$. For the DTW tensor $\mathbf{A}$ (cf. Eqn. (5)), we have $\mathcal{B}_{t,t'} = \{(t+1,t'), (t,t'+1), (t+1,t'+1)\}$, if $0 < t$ and $0 < t'$.

Hence, from Eqn. (8), we get the probability that the process is in state $S_{t,t'}$ at step τ given the complete observation

$$P(z_\tau = S_{t,t'}|o_1^n, {o'}_1^{n'}, \theta) = \frac{\alpha_\tau(t,t')\beta_\tau(t,t')}{P(o_1^n, {o'}_1^{n'}|\theta)}. \tag{11}$$

Any tensor $\mathbf{A}$ satisfying Eqn. (6) is not eligible: for the $\alpha_{t,t'}$ and $\beta_{t,t'}$ recursions to be calculable, one has to impose *linearity*. Basically, $\alpha_{t,t'}$ cannot depend on any $\alpha_{u,v'}$ that is not previously evaluated. The constraint we need to impose is that the time stamps are locally increasing, i.e., if $\alpha_{t,t'}$ depends on any $\alpha_{u,v'}$, then necessarily $[(t < u$ and $t' \leq v')$ or $(t \leq u$ and $t' < v')]$. The same applies for the $\beta_{t,t'}$ recursion.

As an example, Fig. 2 presents the forward backward (FB) matrix $(FB(t,t') = P(S_{t,t'}|o_1^n, {o'}_1^{n'}, \theta))$ corresponding to the alignment of a positive half-wave with a sinus wave. The three areas of likely alignment paths are clearly identified in gray scale colors.

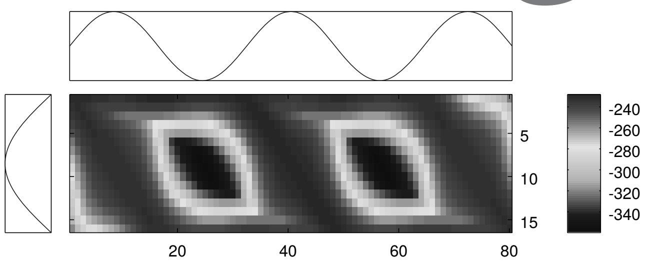


Fig. 2. Forward backward matrix (logarithmic values) for the alignment of a positive halfwave with a sinus wave. The dark gray color color represents high probability states, while the black color represents low probability states.

3.3. Parallel with KRDTW.

A direct parallel exists between KRDTW and the previous Markov process. It follows from the forward equation (9) that

$$\begin{aligned} K_{rdtw}(o_1^k, {o'}_1^l) &= \sum_{i,j} a_{ij,kl} b_{kl} K_{rdtw}(o_1^i, {o'}_1^j) \\ &= \kappa(o_k, {o'}_l) \sum_{i,j} a_{ij,kl} K_{rdtw}(o_1^i, {o'}_1^j), \end{aligned} \tag{12}$$

where $\mathbf{A} = [a_{ij;kl}]$ is defined in Eqn. (5), and $\mathbf{B} = [b_{kl}]$, defined in Eqn. (7), is such that $b_{kl} = e^{-\nu d_E^2(o_k, {o'}_l)}$. Hence, the K_{rdtw} recursion coincides exactly with the forward recursion (Eqn. (9)). Similarly, we can assimilate the backward recursion (Eqn. (10)) to the K_{rdtw} evaluation of the pair of time series obtained by inverting o_1^n and ${o'}_1^{n'}$ along the time axis. Hence, the forward-backward matrix elements (Eqn. (11)) can be directly expressed in terms of the K_{rdtw} recursions.

Furthermore, the corridor function $h(\cdot)$ that occurs in the K_{rdtw} recursion (Eqn. (4)) modifies directly the structure of the transition tensor $\mathbf{A}$ by setting $a_{ij;kl} = 0$ whenever $h(i,j) = 0$ or $h(k,l) = 0$. Neighbor states may be affected also by the normalization that is required to maintain $\mathbf{A}$ stochastic.

3.4. Time elastic centroid estimate of a set of time series.

Let us introduce the marginal probability for the process to be, at step τ, in one of the states of subset $S_{t,\bullet} = \{S_{t,1}, S_{t,2}, \ldots, S_{t,n'}\}$ given the observations o_1^n and ${o'}_1^{n'}$, namely, the probability that the process visits at step τ one of the states of subset $S_{t,\bullet}$, meaning that sample o_t is aligned with the samples of ${o'}_1^{n'}$,

$$\begin{aligned} &P(z_\tau \in S_{t,\bullet}|o_1^n, {o'}_1^{n'}, \theta) \\ &\quad = \sum_{t'} P(z_\tau = S_{t,t'}|o_1^n, {o'}_1^{n'}, \theta) \\ &\quad = \frac{1}{P(o_1^n, {o'}_1^{n'}|\theta)} \sum_{t'} \alpha_\tau(t,t')\beta_\tau(t,t'). \end{aligned} \tag{13}$$

Also, let us consider, for all t and t', the conditional probability of visiting state $S_{t,t'}$ given the two observation sequences, parameter θ and $S_{t,\bullet}$, namely, the probability that o_t and $o'_{t'}$ are aligned given the knowledge that o_t is aligned with one of the samples of ${o'}_1^{n'}$,

$$\begin{aligned} &P(z_\tau = S_{t,t'} | o_1^n, {o'}_1^{n'}, z_\tau \in S_{t,\bullet}, \theta) \\ &\quad = \frac{P(z_\tau = S_{t,t'} | o_1^n, {o'}_1^{n'}, \theta)}{P(z_\tau \in S_{t,\bullet} | o_1^n, {o'}_1^{n'}, \theta)} \\ &\quad = \frac{\alpha_\tau(t,t')\beta_\tau(t,t')}{\sum_{t'} \alpha_\tau(t,t')\beta_\tau(t,t')}. \end{aligned} \tag{14}$$

The previous equality is easily established using Bayes' rule, because $P(z_\tau = S_{t,t'}, z_\tau \in S_{t,\bullet} | o_1^n, {o'}_1^{n'}, \theta) = P(z_\tau = S_{t,t'} | o_1^n, {o'}_1^{n'}, \theta)$.

Hence, for estimating $P(z_\tau = S_{t,t'} | o_1^n, {o'}_1^{n'}, z_\tau \in S_{t,\bullet}, \theta)$, we only need to evaluate the forward ($\alpha_\tau(t,t')$) and backward ($\beta_\tau(t,t')$) recursions, since $P(o_1^n, {o'}_1^{n'} | \theta)$ is eliminated.

We can then define the expectation of the samples of ${o'}_1^{n'}$ that are aligned with sample o_t (given that o_t is aligned) as well as the expectation of the time of occurrence of the samples of ${o'}_1^{n'}$ that are aligned with o_t as follows:

$$\begin{aligned} E(o'|o_t) &\propto \sum_{t'=1}^{n'} {o'}_{t'} P(z_\tau = S_{t,t'} | o_1^n, {o'}_1^{n'}, z_\tau \in S_{t,\bullet}, \theta), \\ E(t'|o_t) &\propto \sum_{t'=1}^{n'} t' P(z_\tau = S_{t,t'} | o_1^n, {o'}_1^{n'}, z_\tau \in S_{t,\bullet}, \theta). \end{aligned} \tag{15}$$

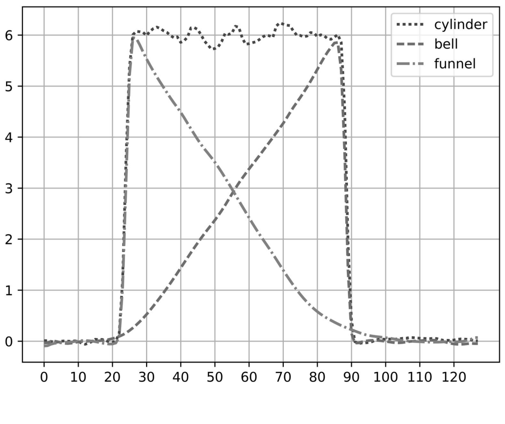


Fig. 3. Centroids obtained for the CBF data set. For the three shapes, the expected start (24) and end (88) time stamps (hence the expected shape duration of 64 frames) are correctly extracted.

The expectations (15) form a basis of our procedure for averaging a set of time series. Let $O = \{{}^k o_1^{n_k}\}_{k=1,\dots,N}$ be a set of time series, and r_1^n a reference time series (r_1^n can be initially set up as the medoid of set O). The centroid estimate of O is defined as the pair $(C, \mathcal{T})$, where C is a time series of length n and $\mathcal{T}$ is the sequence of time stamps associated with the samples of C,

$$\begin{aligned} C_t &= \frac{1}{N} \sum_{k=1}^{N} E({}^k o | r_t) \\ &\propto \frac{1}{N} \sum_{k=1}^{N} \sum_{{}^k t=1}^{n_k} {}^k o_{{}^k t} P(z_\tau = S_{t,{}^k t} | r_1^n, {}^k o_1^{n_k}, z_\tau \in S_{t,\bullet}, \theta), \\ \mathcal{T}_t &= \frac{1}{N} \sum_{k=1}^{N} E({}^k t | r_t) \\ &\propto \frac{1}{N} \sum_{k=1}^{N} \sum_{{}^k t=1}^{n_k} {}^k t P(z_\tau = S_{t,{}^k t} | r_1^n, {}^k o_1^{n_k}, z_\tau \in S_{t,\bullet}, \theta). \end{aligned} \tag{16}$$

Obviously, $(C, \mathcal{T})$ is a non-uniformly sampled time series for which $\mathcal{T}(t)$ is the time stamp associated with observation $C(t)$. $\mathcal{T}(t)$ could be understood as the expected time of occurrence of the expected observation $C(t)$. A uniform re-sampling can straightforwardly be used to get back to a uniformly sampled time series.

The proposed iterative agglomerative algorithm (cf. Fig. 1(b)), called TEKA (time elastic kernel averaging), which provides a refinement of the centroid estimation at each iteration until reaching a (local) optimum is presented as Algorithm 1.

As an example, Fig. 3 presents the obtained time elastic centroid estimates. Using Algorithm 1 with $K = K_{rdtw}$, for the synthetic functions cylinder $c(t)$, bell $b(t)$, funnel $f(t)$ (Saito, 1994) defined as follows:

$$\begin{aligned} c(t) &= (6+\eta)\chi_{[a,b]}(t) + \epsilon(t), \\ b(t) &= (6+\eta)\chi_{[a,b]}(t)\frac{(t-a)}{(b-a)} + \epsilon(t), \\ f(t) &= (6+\eta)\chi_{[a,b]}(t)\frac{(b-t)}{(b-a)} + \epsilon(t), \end{aligned}$$

where $\chi_{[a,b]} = 0$ if $t < a \vee t > b$, 1 if $a \le t \le b$, η and $\epsilon(t)$ are obtained from a standard normal distribution $N(0,1)$, a is an integer obtained from a uniform distribution in $[16, 32]$ and $b - a$ is another integer obtained from another uniform distribution in $[32, 96]$. Hence such shapes are characterized with start and end time stamps of 24 and 88, respectively, and a shape duration of 64 samples. Figure 3 clearly shows that, from a subset of 300 time series (100 for each category), the algorithm has correctly recovered the start and end shape events (hence the expected shape duration) for all three shapes.

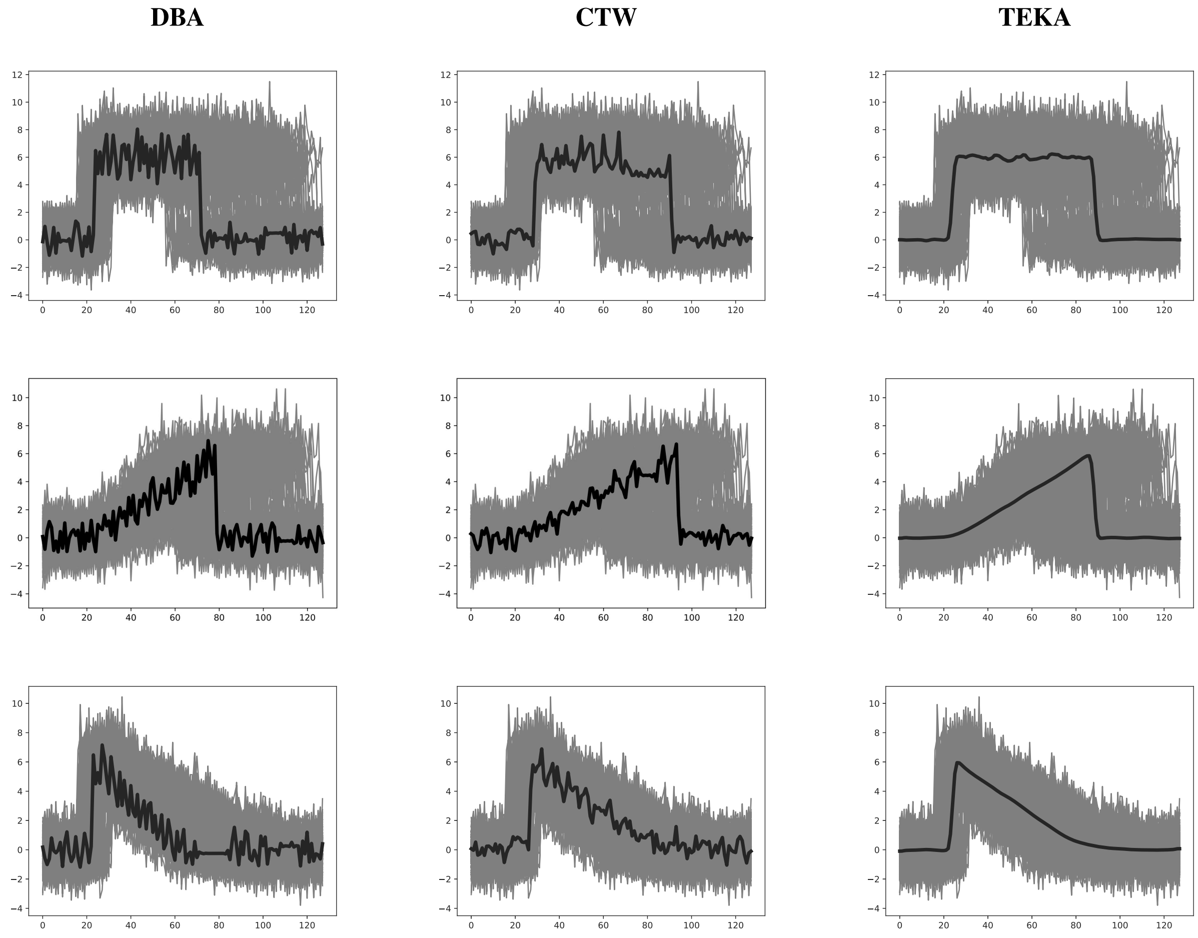


Fig. 4. Centroid estimation for the three categories of the CBF dataset and for the three tested algorithms: DBA (left), CTW (center) TEKA (right). The centroid estimates are indicated as a thick black line superimposed on top of the time series (in light grey) that are averaged.

Figure 4 compares the centroid estimates provided by iterated DBA (Petitjean and Gançarski, 2012), CTW (Zhou and De la Torre, 2009) and TEKA algorithms. For the experiment, the DBA and TEKA algorithms were iterated until convergence. The centroid estimates provided by the TEKA algorithm are much smoother than the ones provided by DBA or CTW. This denoising property, expected from any averaging algorithm, will be addressed in a dedicated experiment (cf. Section 4.3).

3.5. Role of parameter ν. In practice, the selectivity or bandwidth of the local alignment kernels (that is, controlled by parameter ν) has to be adapted according to the lengths of the time series. If the time series are long, then ν should be reduced to maintain the calculability of the forward-backward matrices, while the local selectivity decreases. Hence, more alignment paths are likely and more sample pairs participate in the calculation of the average such that local details are filtered out by the averaging. Conversely, if the time series are short, ν can be increased, hence fewer sample pairs participate in the calculation of the average, and details can be preserved.

Hence parameter ν controls the smoothing of the centroid, as exemplified in Fig. 5 for the CBF dataset. Too small a value of ν will filter out high frequencies and the TEKA algorithm acts as a low-pass filter. On the top sub-figure, for $\nu = 0.001$, the cylinder, bell and funnel shapes are distorted, in particular near the discontinuities of the functions. When ν is too high, the probability estimates given in Eqn. (14) are vanishing and the TEKA centroid calculation is not reliable anymore. This is what happens for $\nu = 10$ in the bottom sub-figure, in which the centroids for the bell and funnel functions converge (in the context of our current implementation) toward the constant functions that correspond to the mean values of the shapes. For a “correct” ν value, the additive noise can be filtered out while keeping neatly the discontinuities that are present in the shapes. This is the case for the CBF data

Algorithm 1. Iterative time elastic kernel averaging (TEKA) of a set of time series.

```
Let $K$ be a similarity time elastic kernel for time series satisfying Eqn. (12)
Let $O$ be a set of time series of $d$ dimensional samples
Let $C_0$ be an initial centroid estimate (e.g., the medoid of $O$) of length $n$
Let $\mathcal{T}$ and $\mathcal{T}_0$ be two sequences of time stamps of length $n$ initialized with zero values
Let $MeanK_0 = 0$ and $MeanK$ be two double values;
repeat
    $C = C_0, \mathcal{T} = \mathcal{T}_0, MeanK = MeanK_0$;
    Evaluate $C_0$ and $\mathcal{T}_0$ according to Eqn. (16)
    //Average similarity between $C_0$ and elements of $O$
    $MeanK_0 = \frac{1}{|O|}\sum_{o \in O} K(C_0, o)$
until $MeanK \leq MeanK_0$
$(C, \mathcal{T})$ is the centroid estimation
Finally, uniformly re-sample $C$ using the time stamps $\mathcal{T}$
```

set when $\nu = .1$, as depicted in the sub-figure located at the center of Fig. 5.

The upper bound for ν can be experimentally (and automatically) tuned to ensure that, for a given data set, the numerical vanishing of the probability estimates given in Eqn. (14) is avoided. The lower bound for ν is 0^+. Finding the "correct" value for ν within these two bounds is unfortunately dependent on the application. However, in a time series classification framework, ν can be automatically optimized using a cross-validation or a leave-one-out procedure, for instance. In a clustering framework, the expertise of the practitioner (the knowledge of the physical process behind the production of the time series) is in general required to determine an acceptable ν value. In some cases, few utterances of the clean signal are available and can be used to adjust the ν value, similarly to a procedure used for adjusting the parameters of a band-pass filter to improve or optimize a signal to noise ratio.

3.6. Computational complexity.

TEKA has intrinsically the same algorithmic complexity as the DBA algorithm; basically $O(L^2)$ for each pairwise averaging, where L is the average length of the time series. Nevertheless, computationally speaking, the TEKA algorithm is slightly more costly mainly because of two reasons:

- the FB matrix induces a factor three in complexity because of the reverse alignment and the multiplication term by term of the forward and backward matrices;
- the exponential terms that enter into the computation of KRDTW (Eqn. (4)) are costly; basically, $O(M(n)n^{1/2})$, where $M(n)$ is the cost of the floating point multiplication, and n is the number of digits. This induces another factor 2 or 3, depending on the chosen floating point precision.

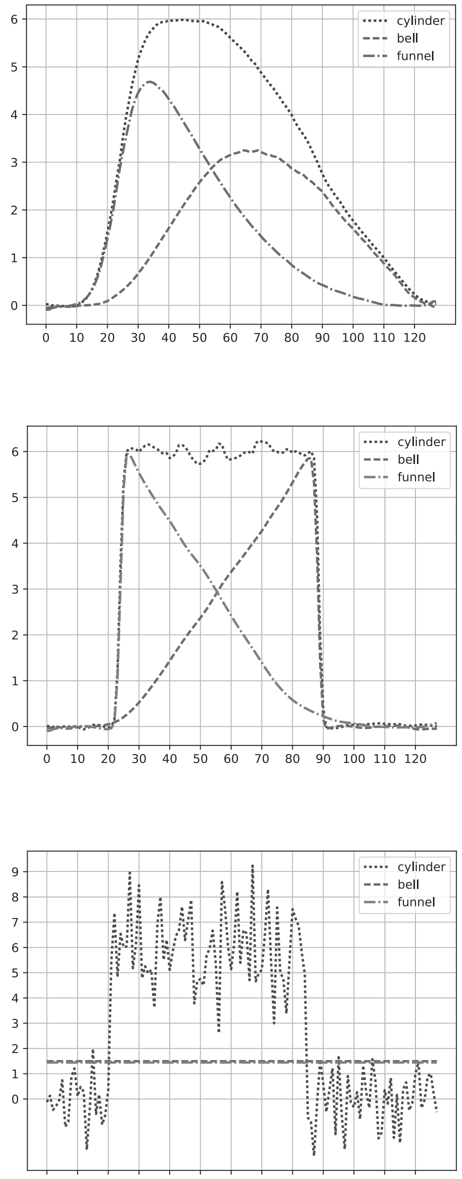


Fig. 5. Centroid estimation for the three categories of the CBF dataset with $\nu = .001$ (top), $\nu = .1$ (middle) and $\nu = 10$ (bottom).

The overall algorithmic cost for averaging a set of N time series of average length L with an average number of iterations I is, for the two algorithms, $O(I \cdot N \cdot L^2)$.

Some optimization is indeed possible, in particular replacing the exponential function by another local kernel easier to compute is an important source of algorithmic simplification. We do not address further this issue in this paper and let it stand as a perspective.

4. Experiments

The two first proposed experiments aim at demonstrating the benefits of using time elastic centroids in a data reduction paradigm: 1-NC/NM (first near centroid or medoid) classification for the first one, and isolated gesture recognition for the second one using 1-NC/NM and SVM classifiers in conjunction with the KRDTW kernel. The third experiment explores the noise reduction angle brought by time elastic centroids.

4.1. 1-Nearest centroid/medoid classification. The purpose of this experiment is to evaluate the effectiveness of the proposed time elastic averaging method (TEKA) against a triple baseline. The first baseline allows us to compare centroid-based with medoid-based approaches. The second and third baselines are provided by the DBA (Petitjean and Gançarski, 2012) and CTW (Zhou and De la Torre, 2009) algorithms (thanks to the implementation proposed by the authors), currently considered a state of the art methods to average a set of sequences consistently with DTW. We have tested the CTW averaging with a 1-NC-DTW (CTW1) and a 1-NC-KRDTW (CTW2) classifier to highlight the impact of the selected similarity measure.

For this purpose, we empirically evaluate the effectiveness of the methods using a first nearest centroid/medoid (1-NC/NM) classification task on a set of time series derived from widely diverse fields of application. The task consists in representing each category contained in a training data set by estimating its medoid or centroid and then evaluating the error rate of a 1-NC classifier on an independent testing data set. Hence, the classification rule consists of assigning to the tested time series the category which corresponds to the closest (or most similar) medoid or centroid according to the DTW measure for the DTW medoid (DTW-M) and the DBA and CTW centroids (CTW1) or to the KRDTW measure for KRDTW medoid (KRDTW-M), CTW (CTW2) and TEKA centroids.

In the work of Petitjean *et al.* (2014) a generalized k-NC task is described. The authors demonstrate that, by selecting the appropriate number k of centroids (using DBA and k-means), they achieve, without loss, a 70% speed-up on average, compared to the original k-nearest neighbor task. Although, in general, the classification accuracy is improved when several centroids are used to represent the training datasets, our main purpose is to highlight and amplify the discrimination between time series averaging methods: this is why we stick here to the 1-NC task.

A collection of 45 heterogeneous data sets is used to assess the proposed algorithms. The collection includes synthetic and real data sets, as well as univariate and multivariate time series. These data sets are distributed as follows:

- 42 of these data sets are available at the UCR repository (Keogh *et al.*, 2006). Basically, we used all the data sets except for *StarLightCurves*, *Non-Invasive Fetal ECG Thorax1* and *Non-Invasive Fetal ECG Thorax2*. Although these last three data sets are still tractable, their computational cost is high because of their size and the length of the time series they contain. All these data sets are composed of scalar time series.
- One data set, *uWaveGestureLibrary_3D* was constructed from the *uWaveGestureLibrary_X—Y—Z* scalar data sets to compose a new set of multivariate (3D) time series.
- One data set, *CharTrajTT*, is available at the UCI repository (Lichman, 2013) under the name *Character Trajectories Data Set*. This data set contains multivariate (3D) time series and is divided into two equal sized data sets (*TRAIN* and *TEST*) for the experiment.
- The last data set, *PWM2*, which stands for *Pulse Width Modulation* (Marteau, 2007), was specifically defined to demonstrate a weakness in the dynamic time warping pseudo distance. This data set is composed of synthetic scalar time series.

For each dataset, a training subset (*TRAIN*) is defined as well as an independent testing subset (*TEST*). We use the training sets to extract single medoids or centroid estimates for each of the categories defined in the data sets.

Furthermore, for KRDTW-M, CTW2 and TEKA, the ν parameter is optimized using a *leave one out* (LOO) procedure carried out on the *TRAIN* data sets. The ν value is selected within the discrete set $\{.01, .05, .1, .25, .5, .75, 1, 2, 5, 10, 15, 20, 25, 50, 100\}$. The value that minimizes the LOO classification error rate on the *TRAIN* data is then used to provide the error rates that are estimated on the *TEST* data.

The classification results are given in Table 1. It can be seen from this experiment that

1. centroid-based methods outperform medoid-based ones: DBA and CTW (CTW2) yield lower error rates compared to DTW-M, as does TEKA compared with KRDTW-M and DTW-M;
2. CTW pairs much better with KRDTW (CTW2 outperforms CTW1);

Table 1. Comparative study using the UCR and UCI data sets: classification error rates evaluated on the TEST data set (in %) obtained using the first nearest neighbour classification rule for DTW-M, KRDTW-M, (medoids), DBA, CTW1, CTW2 and TEKA (centroids). A single medoid/centroid extracted from the training data set represents each category.

Dataset	# Cat | L	DTW-M	DBA	CTW1	CTW2	KRDTW-M	TEKA
Synthetic_Control	6|60	3.00	**2.00**	19.00	3.33	3.33	2.33
Gun_Point	2|150	44.00	32.00	54.67	**25.33**	52.00	27.33
CBF	3|128	7.89	5.33	34.22	3.55	8.11	**3.33**
Face_(all)	14|131	25.21	18.05	34.38	27.93	20.53	**13.61**
OSU_Leaf	6|427	64.05	56.20	64.05	57.02	53.31	**50.82**
Swedish_Leaf	15|128	38.56	30.08	32	25.76	31.36	**22.08**
50Words	50|270	48.13	41.32	48.57	36.48	23.40	**19.78**
Trace	4|275	**5.00**	7.00	6.00	18	23.00	16.00
Two_Patterns	4|128	1.83	1.18	26.75	37.75	1.17	**1.10**
Wafer	2|152	64.23	33.89	37.83	33.27	43.92	**8.38**
Face_(four)	4|350	12.50	13.64	19.32	15.91	17.05	**10.23**
Lightning-2	2|637	34.43	37.70	37.70	**29.51**	**29.51**	**29.51**
Lightning-7	7|319	27.40	27.40	41.10	38.35	19.18	**16.44**
ECG200	2|96	32.00	28.00	27.00	**25**	29.00	26.00
Adiac	37|176	57.54	52.69	54.73	34.78	40.67	**32.22**
Yoga	2|426	47.67	47.87	53.56	48.97	47.53	**44.90**
Fish	7|463	38.86	30.29	39.42	22.28	20.57	**14.28**
Beef	5|470	60.00	53.33	53.33	**50**	53.33	**50**
Coffee	2|286	57.14	32.14	32.14	**28.57**	32.14	32.14
OliveOil	4|570	26.67	**16.67**	13.33	23.33	30	**16.67**
CinC_ECG_torso	4|1639	74.71	53.55	73.33	42.90	66.67	**33.04**
ChlorineConcentration	3|166	65.96	68.15	67.40	67.97	65.65	**64.97**
DiatomSizeReduction	4|345	22.88	5.88	5.23	**2.61**	11.11	2.94
ECGFiveDays	2|136	47.50	30.20	34.49	13.47	**11.38**	16.37
FacesUCR	14|131	27.95	18.44	32.20	21.66	20.73	**12.19**
Haptics	5|1092	68.18	64.61	58.77	57.47	63.64	**53.57**
InlineSkate	7|1882	78.55	76.55	81.64	82.18	78.36	**75.09**
ItalyPowerDemand	2|24	31.68	20.99	15.84	9.33	**5.05**	6.61
MALLAT	8|1024	6.95	6.10	5.24	**3.33**	6.87	3.66
MedicalImages	10|99	67.76	58.42	58.29	59.34	**57.24**	59.60
MoteStrain	2|84	15.10	13.18	19.01	15.33	12.70	**9.35**
SonyAIBORobot_SurfaceII	2|65	26.34	21.09	20.57	**17.52**	26.230	19.30
SonyAIBORobot_Surface	2|70	38.10	19.47	14.48	**9.31**	39.77	17.95
Symbols	6|398	7.64	4.42	22.31	20.70	**3.92**	4.02
TwoLeadECG	2|82	24.14	**13.17**	20.37	19.23	27.04	18.96
WordsSynonyms	25|270	70.85	64.26	78.84	63.32	64.26	**56.11**
Cricket_X	12|300	67.69	**52.82**	78.46	73.85	61.79	**52.82**
Cricket_Y	12|300	68.97	52.82	69.74	65.64	**46.92**	50.25
Cricket_Z	12|300	73.59	**48.97**	78.21	64.36	56.67	51.79
uWaveGestureLibrary_X	8|315	38.97	33.08	37.33	34.61	34.34	**32.18**
uWaveGestureLibrary_Y	8|315	49.30	44.44	45.42	41.99	42.18	**39.64**
uWaveGestureLibrary_Z	8|315	47.40	**39.25**	47.65	39.36	41.96	39.97
PWM2	3|128	43.00	35.00	63.66	6.33	21.00	**4.33**
uWaveGestureLibrary_3D	8|315	10.11	**5.61**	9.35	7.68	13.74	7.73
CharTrajTT_3D	20|178	11.026	9.58	13.45	15.05	6.93	**4.99**
# Best Scores	–	1	7	0	9	6	**27**
# Uniquely Best Scores	–	1	5	0	7	5	**23**
Average rank	–	4.56	2.87	4.62	2.97	3.22	**1.6**

3. TEKA outperforms DBA (under the same experimental conditions) and CTW.

The average ranking for all six tested methods, which supports our preliminary conclusion, is given at the bottom of Table 1.

In Table 2 we report the p-values for each pair of the tested algorithms using a Wilcoxon signed-rank test. The null hypothesis is that, for a tested pair of classifiers, the difference between classification error rates obtained on the 45 data sets follows a symmetric distribution around zero. With a $.05$ significance level, the p-values that lead to rejecting the null hypothesis are shown in boldface in the table. This analysis confirms our previous study of the classification results. We observe that centroid-based approaches perform significantly better than medoid-based ones. Furthermore, KRDTW-M appears to be significantly better than DTW-M.

Furthermore, TEKA is evaluated as significantly better than DBA and CTW2 in this experiment. Note also that DBA does not seem to perform significantly better than KRDTW-M or CTW2, and that CTW1 performed similarly to DTW-M and poorly compared to the other centroid methods. Hence, this confirms out that CTW method seems to pair well with the KRDTW measure but poorly with the DTW one.

4.2. Instance set reduction. In this second experiment, we address an application that consists in summarizing subsets of training time series to speed up an isolated gesture recognition process.

The data set that we consider (Ghouaiel *et al.*, 2017) enables us to explore the hand-shape and the upper body movement using 3D positions of skeletal joints captured using a Microsoft Kinect 2 sensor. 20 subjects were selected (15 males and 5 females) to perform in front of the sensor (at a three-meter distance) the 6 selected NATOPS gestures. Each subject repeated each gesture three times. Hence the isolated gesture dataset is composed of 360 gesture utterances that have been manually segmented to a fixed length of 51 frames.[1].

An excerpt of this multivariate time series database is shown in Fig. 6. The 3D positions for the thumbs, hand extremities, elbows and shoulders are shown as a function of time for the *Lock Wings* gesture.

To evaluate this task, we performed a subject cross-validation experiment consisting of 100 tests: for each test, 10 subjects were randomly drawn among 20 for training and the remaining 10 subjects were retained for testing. The 1-NN/NC (our baselines) and SVM classifiers are evaluated, with or without summarizing the subsets composed of the three repetitions performed by each subjects using a single centroid (DBA, CTW, TEKA) or medoid (KRDTW-M). The parameter ν of the KRDTW kernel as well as the SVM meta parameter (RBF bandwidth σ and C) are optimized using a leave one subject procedure on the training dataset. The kernels $\exp(-\mathrm{DTW}(\cdot,\cdot)/\sigma)$ and $\exp(-\mathrm{KRDTW}(\cdot,\cdot)/\sigma)$ are used respectively in the SVM DTW and SVM KRDTW classifiers.

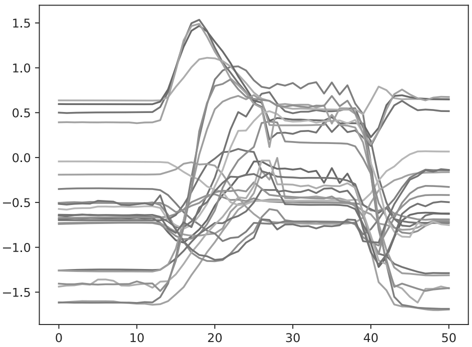


Fig. 6. Excerpt of the gesture database: 3D positions evolving with time for two hand extremities; two thumbs and two elbows are shown.

Table 3 gives the assessment measures (ERR: average error rate, PRE: macro average precision, REC: macro average recall and $F_1 = 2 \times \frac{\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}}$) for the isolated gesture classification task. In addition, the number of reference instances used by the 1-NN/NC classifiers or the number of support vectors exploited by the SVM ($\overline{\#\text{Ref}}$ column in the table) are reported to demonstrate the data reduction that is induced by the methods in the training sets.

The results show that the DTW measure does not fit well with the SVM compared with KRDTW: the error rate or the F1 score are about 9% higher or lower for the isolated gesture task. Hence, to compare the DBA, CTW and TEKA centroids using an SVM classification, the KRDTW kernel was used. When employing the centroids (SVM KRDTW-DBA, SVM KRDTW-CTW, SVM KRDTW-TEKA) or medoids (SVM KRDTW-M), the error rate or F_1 score increases or decreases only by around 2.5% and 2% compared with SVM-KRDTW, which achieves the best scores. Meanwhile, the number of support vectors exploited by the SVM drops by a factor of 2, leading to an expected speed-up of 2. Compared with 1-NN classification without centroids, SVM KRDTW with centroids achieves much better performance, with an expected speed-up of 4 ($\sim$ 50 support vectors comparatively to 180 gesture instances). This demonstrates the capacity of centroid methods to reduce significantly the size of the training sets while

[1] This dataset is available at `https://github.com/pfmarteau/IGR_Kinect_DB`.

Table 2. Wilcoxon signed-rank test of pairwise accuracy differences for 1-NC/NM classifiers carried out on the 45 datasets.

Method	KRDTW-M	DBA	CTW1	CTW2	TEKA
DTW-M	**p<.0001**	**p<.0001**	0.638	**0.0002**	**p<.0001**
KRDTW-M	–	0.395	**0.0004**	0.5261	**p<.0001**
DBA	–	–	**p<.0001**	0.8214	**p<.0001**
CTW1	–	–	–	**p<.0001**	**p<.0001**
CTW2	–	–	–	–	**p<.0001**

Table 3. Assessment measures (ERR: error rate, PRE: precision, REC: recall and $\mathbf{F_1}$ score) for the isolated gestures recognition. $\overline{\#\text{Ref}}$ is the number of training gestures for the 1-NN/NC classifiers and the mean number of support vectors for the SVM classifiers.

Method	ERR mean ‖ std	PRE	REC	F_1	$\overline{\#\text{Ref}}$
1-NN DTW	.134 ‖ .012	.869	.866	0.867	180
1-NN KRDTW	**.128** ‖ .016	.876	.972	.874	180
1-NC DTW-DBA	.136 ‖ .014	.868	.864	.866	**60**
1-NC KRDTW-CTW	.135 ‖ .016	.871	.865	.868	**60**
1-NC KRDTW-TEKA	**.133** ‖ .014	.871	.867	.869	**60**
SVM DTW	.146 ‖ .015	.871	.854	.862	164.97
SVM KRDTW	**.051** ‖ .015	.952	.949	.951	**103.10**
SVM KRDTW-M	.087 ‖ .02	.92.9	.92.6	.92.7	47.62
SVM KRDTW-DBA	.080 ‖ .017	.935	.931	.931	**46.74**
SVM KRDTW-CTW	.085 ‖ .021	.933	.927	.930	50.12
SVM KRDTW-TEKA	**.079** ‖ .019	.937	.933	.935	47.45

Table 4. Wilcoxon signed-rank test of pairwise accuracy differences for 1-NN/NC classifiers. The DTW and KRDTW methods exploit the entire training sets while the other methods only use one centroid for each subject and each gesture label.

Method	1-NN KRDTW	1-NC DBA	1-NC CTW	1-NC TEKA
1-NN DTW	**p<.0001**	0.140	0.886	0.371
1-NN KRDTW	–	**p<.0001**	**0.026**	0.087
1-NC DBA	–	–	0.281	**0.006**
1-NC CTW	–	–	–	0.199

Table 5. Wilcoxon signed-rank test of pairwise accuracy differences for SVM classifiers. The DTW and KRDTW methods exploit the entire training sets while the other methods only use one centroid for each subject and each gesture label.

Method	SVM KRDTW	SVM KRDTW-M	SVM DBA	SVM CTW	SVM TEKA
SVM DTW	**p<.0001**	**p<.0001**	**p<.0001**	**p<.0001**	**p<.0001**
SVM KRDTW	–	**p<.0001**	**p<.0001**	**p<.0001**	**p<.0001**
SVM KRDTW-M	–	–	**0.002**	0.57	**0.0002**
SVM DBA	–	–	–	0.107	0.339
SVM CTW	–	–	–	–	**0.013**

maintaining a very similar level of accuracy.

In greater detail, TEKA is the centroid-based method that achieves the lowest error rates for the two classification tasks, while DBA is the centroid-based method that exploits the fewest support vectors (46.5).

Tables 4 and 5 give the p-values for the Wilcoxon signed-rank tests. With the same null hypothesis as above (the difference between the error rates follows a symmetric distribution around zero) and with a .05 significance level, the p-values that lead to rejecting the null hypothesis are presented in boldface in the tables.

From Table 4 we note that 1NN-KRDTW (which exploits the full training set) performs significantly better than 1NN DTW, 1-NC DTW-DBA and 1-NC KRDTW-CTW, but not significantly better than 1-NC KRDTW-TEKA. Conversely, 1-NC KRDTW-TEKA performs significantly better than 1-NC DTW-DBA but not significantly better that 1-NC KRDTW-CTW. Similarly, from Table 5 we observe that SVM KRDTW, which exploits the full training set, performs significantly better than all centroid or medoid based methods. Also, SVM KRDTW-TEKA performs significantly better than SVM KRDTW-CTW but not significantly better than SVM KRDTW-DBA. Finally, SVM KRDTW-TEKA and SVM KRDTW-DBA outperform the medoid based method (SVM KRDTW-M) but not SVM KRDTW-CTW.

If the three centroid methods show rather close accuracies on this experiment, TEKA is significantly better than DBA on the 1NC task and significantly better than CTW on the SVM task.

4.3. Denoising experiment. To demonstrate the utility of centroid based methods for denoising data, we construct a demonstrative synthetic experiment that provides some insights. The test is based on the following 2D periodic signal:

$$X_k(t) = \left(A_k + B_k \sum_{i=1}^{\infty} \delta(t - \frac{2\pi i}{6\omega_k}) \right) \cos(\omega_k t + \phi_k), \tag{17}$$

$$Y_k(t) = \left(A_k + B_k \sum_{i=1}^{\infty} \delta(t - \frac{2\pi i}{6\omega_k}) \right) \sin(\omega_k t + \phi_k),$$

where $A_k = A_0 + a_k$, $B_k = (A_0 + 5) + b_k$ and $\omega_k = \omega_0 + w_k$, A_0 and ω_0 are constant, and a_k, b_k, ω_k, ϕ_k are small perturbations in the amplitude, frequency and phase, respectively, and randomly drawn from $a_k \in [0, A_0/10]$, $b_k \in [0, A_0/10]$, $\omega_k \in [-\omega_0/6.67, \omega_0/6.67]$, $\phi_k \in [-\omega_0/10, \omega_0/10]$.

In practice we adopted the following setting: $f_0 = \omega_o/(2\pi) = 20\,\text{Hz}$, and $A_0 = 1$. We then center and normalize this 2D signal to get $(\tilde{X}_k(t), \tilde{Y}_k(t))$ corresponding to the plots given in the top of Fig. 7. The log-power spectrum of the $\tilde{X}_k$ component shows the Dirac spike located at $f_0 = 20$ Hz (corresponding to the sine component), and the convolution of this spike with a Dirac comb in the frequency domain that results in pairs of Dirac spikes symmetrically located (±20 Hz) around multiples of $6f_0$, namely, 120 Hz, 240 Hz, etc. This shows that the signal is characterized by an infinite spectrum.

We then consider noise utterances $\epsilon_k(t)$ with zero mean and unit variance, added to each instances of the 2D signal:

$$x_k(t) = \tilde{X}_k(t) + \epsilon_k(t),$$
$$y_k(t) = \tilde{Y}_k(t) + \epsilon_k(t),$$

leading to a signal to noise ratio of 0 dB. An example of such a noisy instance is given in the bottom of Fig. 7. Because of the scattering of the random components of the signal in a wide spectral band, traditional noise reduction techniques, such as those presented by Hassan and Anwar (2010), for instance, will not allow us to recover the signal properly.

The task consists in reducing the noise as far as possible to recover the 2D shape of the noise free signal from a small set of noisy instances $\{(x_k, y_k)\}_{k=1,\ldots,8}$ containing two "periods" of the clean signal. Figure 8 presents the centroid shapes obtained using, from left to right, the Euclidean, DBA, CTW and TEKA methods, respectively. We can see that the Euclidean centroid retrieves partially the low frequency sine component without properly sorting out the spikes components, while DBA more accurately retrieves the spikes, although without achieving to suppress the low frequency noise around the sine component. The CTW centroid appears to be in between and partially reduces the low frequency noise and extracts the spikes. TEKA achieves the best retrieval of the sine and spikes components that are better timely and spatially separated. The spectral analysis presented in Fig. 8 (top) gives further insight: for DBA and CTW centroids, (top center sub-figures), the series of pairs of Dirac spikes (in dotted gray) are still hidden into the noise level (black curve), while being much more separated from the noise for the TEKA centroid, as shown in the top right side sub-figure.

Moreover, if we take the clean shapes as ground truth, the signal to noise ratio (SNR) gains estimated from the log-power spectra (to get rid of the phase) is 0 dB for the noisy shapes, 1.58 dB for the Euclidean centroid, 1.17 dB for the DBA centroid, 1.57 dB for the CTW centroid and 3.88 dB for the TEKA centroid. Note that in the calculation of the SNR, preserving the spikes has a lower impact compared to preserving the low frequency sine wave, which explains why the SNR values obtained by the DBA and CTW centroid are lower than for the Euclidean one.

In terms of noise reduction, this experiment demonstrates the ability of the TEKA centroid to better recover, from few noisy utterances, a signal whose components are scattered in a wide band spectrum. Indeed, if the noise level increases, the quality of the denoising will be reduced.

4.4. Discussion. We believe that the noise filtering ability of TEKA is mainly due to the averaging technique described in Eqn. (16), which aggregates many plausible

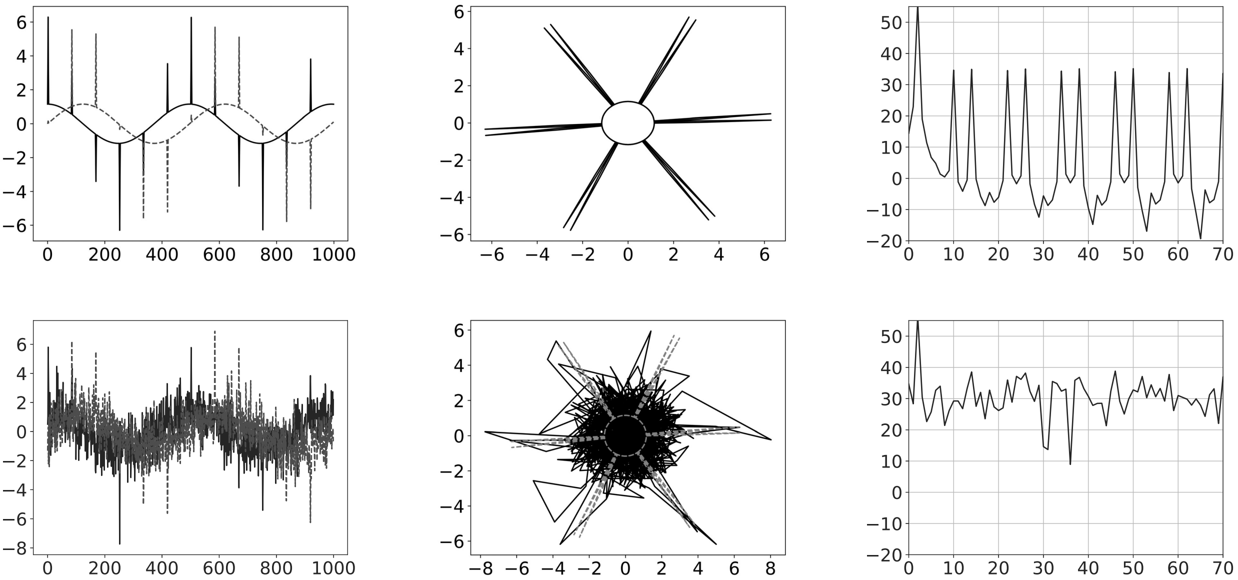


Fig. 7. Waveforms, 2D shape and power spectra for the clean synthetic signal (top) and the noisy synthetic signal (bottom).

alignments between samples (instead of a best one) while averaging also the time of occurrence of the samples, in particular those corresponding to the expected pattern location and duration such as the CBF shapes or the spike locations in the third experiment. This ability is also likely to explain the best accuracy results obtained by TEKA compared to the state of the art methods, CTW and DBA.

Furthermore, it seems that the KRDTW measure is more adapted to match centroids than DTW. Here again, handling several good to best alignments rather than a single optimal one allows matching the centroids in many ways that are averaged by the measure. This has been verified for CTW in 1-NC classification tasks and is true for TEKA and DBA as well.

The main limitation in exploiting TEKA (and KRDTW) is the tuning of the parameter ν that controls the selectivity of the local kernel. Note that ν is dependent on the length of the time series and needs to be adapted to the task itself. Basically, if ν is too small, TEKA will filter out high frequency events just as a moving average filter. Conversely, if ν is too high, the computation of the products of local probabilities along the alignment paths will bear some loss of significance in terms of the numerical calculation. Despite this tuning requirement, the three experiments, that we have carried out in this study, demonstrate its applicability and usefulness.

5. Conclusion

In this paper, we addressed the problem of averaging a set of time series in the context of a time elastic distance measure such as dynamic time warping. The new perspective provided by the kernelization of the elastic distance allows a re-interpretation of pairwise kernel alignment matrices as the result of a forward-backward procedure applied on the states of equivalent stochastic alignment automata. Following this re-interpretation, we proposed a new algorithm, TEKA, based on an iterative agglomerative heuristic method that allows efficient computing good solutions to the multi-alignment of time series. This algorithm exhibits quite interesting denoising capabilities, which enlarges the area of its potential applications.

We reported extensive experiments carried out on synthetic and real data sets, containing univariate but also multivariate time series. Our results show that centroid-based methods significantly outperform medoid-based ones in the context of a first nearest neighbor and SVM classification tasks. More strikingly, the TEKA algorithm, which integrates joint averaging in the sample space and along the time axis, is significantly better than the state-of-the art DBA and CTW algorithms, with a similar algorithmic complexity. It enables robust training set reduction, which was experimented on an isolated gesture recognition task. Finally, we developed a dedicated synthetic test to demonstrate the denoising capability of our algorithm, a property that is not supported at the same level by the other time-elastic centroid methods on this test.

Acknowledgment

The author thanks the French Ministry of Research, the Brittany Region and the European Regional Development

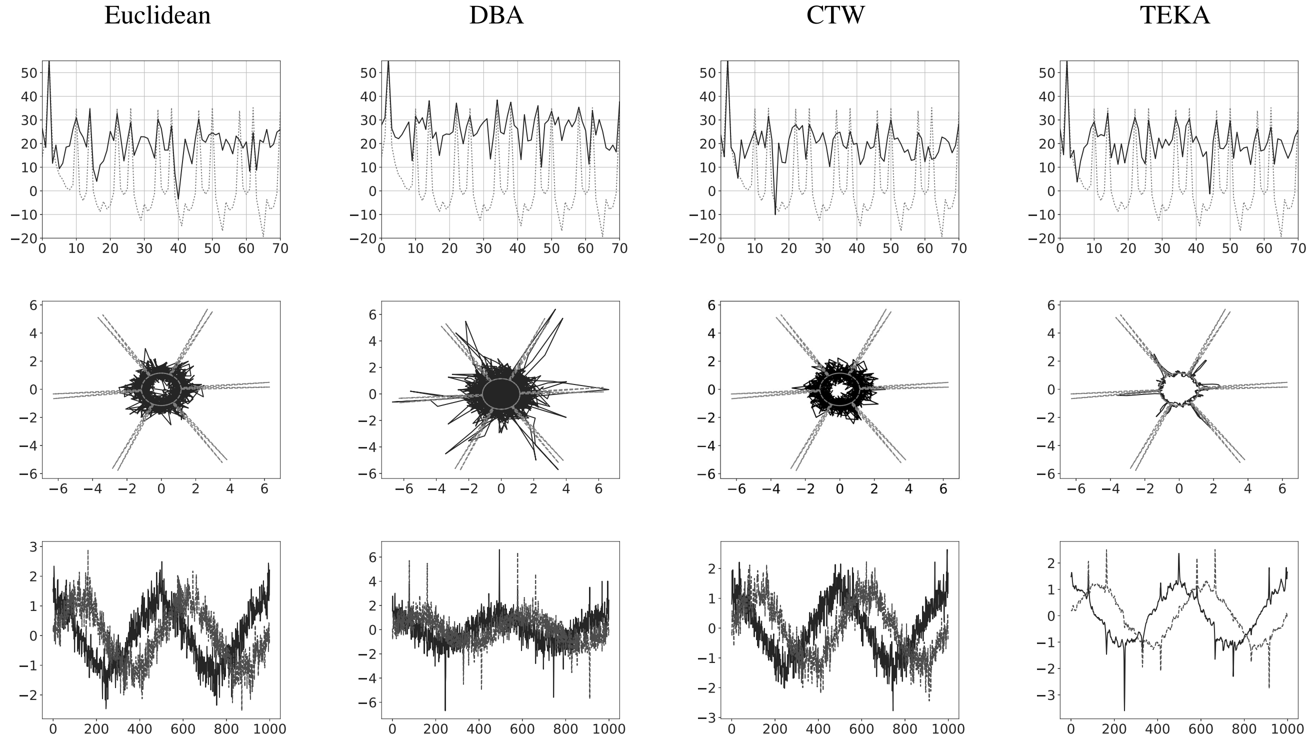


Fig. 8. Centroids obtained from a set of height noisy instances $\{(x_k, y_k)\}_{k=1,\ldots,8}$ for the Euclidean, DBA, CTW and TEKA averaging methods. The log power spectra in dB (top), the 2D shape (center) and x,y waveforms (bottom) are shown.

Fund, which partially funded this research. He also acknowledges the creators of the UCR and UCI data repositories for providing the datasets used in this study.

References

Abdulla, W., Chow, D. and Sin, G. (2003). Cross-words reference template for DTW-based speech recognition systems, *Conference on Convergent Technologies for the Asia-Pacific Region TENCON 2003, Bangalore, India*, Vol. 4, pp. 1576–1579.

Carrillo, H. and Lipman, D. (1988). The multiple sequence alignment problem in biology, *SIAM Journal on Applied Mathematics* **48**(5): 1073–1082.

Chen, L. and Ng, R. (2004). On the marriage of Lp-norms and edit distance, *Proceedings of the 30th International Conference on Very Large Data Bases, VLDB'04, Toronto, Canada*, pp. 792–803.

Chudova, D., Gaffney, S. and Smyth, P. (2003). Probabilistic models for joint clustering and time-warping of multidimensional curves, *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence, UAI'03, San Francisco, CA, USA*, pp. 134–141.

Cuturi, M., Vert, J.-P., Birkenes, O. and Matsui, T. (2007). A kernel for time series based on global alignments, *IEEE ICASSP 2007, Honolulu, HI, USA*, Vol. 2, pp. II-413–II-416.

Fasman, K.H. and Salzberg, S.L. (1998). An introduction to biological sequence analysis, *in* S.L. Salzberg *et al.*, *Computational Methods in Molecular Biology*, Elsevier, Amsterdam, pp. 21–42.

Fréchet, M. (1906). *Sur quelques points du calcul fonctionnel*, Thèse, Faculté des sciences de Paris, Paris.

Ghouaiel, N., Marteau, P.-F. and Dupont, M. (2017). Continuous pattern detection and recognition in stream—a benchmark for online gesture recognition, *International Journal of Applied Pattern Recognition* **4**(2): 146–160.

Gupta, L., Molfese, D., Tammana, R. and Simos, P. (1996). Nonlinear alignment and averaging for estimating the evoked potential, *IEEE Transactions on Biomedical Engineering* **43**(4): 348–356.

Gupta, M., Gao, J., Aggarwal, C.C. and Han, J. (2014). Outlier detection for temporal data: A survey, *IEEE Transactions on Knowledge and Data Engineering* **26**(9): 2250–2267.

Hassan, U. and Anwar, M.S. (2010). Reducing noise by repetition: Introduction to signal averaging, *European Journal of Physics* **31**(3): 453.

Hautamaki, V., Nykanen, P. and Franti, P. (2008). Time-series clustering by approximate prototypes, *19th International Conference on Pattern Recognition, ICPR 2008, Tampa, FL, USA*, pp. 1–4.

Juang, B. (1985). On the hidden Markov model and dynamic time warping for speech recognition—A unified view, *AT&T Bell Laboratories Technical Journal* **63**(7): 1213–1242.

Just, W. and Just, W. (1999). Computational complexity of multiple sequence alignment with SP-score, *Journal of Computational Biology* **8**(6): 615–623.

Kaiser, R. and Knight, W. (1979). Digital signal averaging, *Journal of Magnetic Resonance (1969)* **36**(2): 215–220.

Keogh, E.J., Xi, X., Wei, L. and Ratanamahatana, C. (2006). The UCR time series classification-clustering datasets, *Repository*, `http://www.cs.ucr.edu/~eamonn/time_series_data/`.

Lichman, M. (2013). UCI Machine Learning Repository, `http://archive.ics.uci.edu/ml`.

Marteau, P.-F. (2007). Pulse width modulation data sets, `http://people.irisa.fr/Pierre-Francois.Marteau/PWM/`.

Marteau, P.-F. (2009). Time warp edit distance with stiffness adjustment for time series matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31**(2): 306–318.

Marteau, P.-F. and Gibet, S. (2014). On recursive edit distance kernels with application to time series classification, *IEEE Transactions on Neural Networks and Learning Systems* **26**(6): 1121–1133.

Nakagawa, S. and Nakanishi, H. (1989). Speaker-independent English consonant and Japanese word recognition by a stochastic dynamic time warping method, *Journal of Institution of Electronics and Telecommunication Engineers* **34**(1): 87–95.

Niennattrakul, V. and Ratanamahatana, C. (2007). Inaccuracies of shape averaging method using dynamic time warping for time series data, *in* Y. Shi *et al.* (Eds.), *Computational Science—ICCS 2007*, Lecture Notes in Computer Science, Vol. 4487, Springer, Berlin/Heidelberg, pp. 513–520.

Niennattrakul, V. and Ratanamahatana, C. (2009). Shape averaging under time warping, *6th International Conference on Electronics, Computer, Telecommunications and Information Technology, ECTI-CON 2009, Pattaya, Chonburi, Thailand*, Vol. 02, pp. 626–629.

Petitjean, F., Forestier, G., Webb, G., Nicholson, A., Chen, Y. and Keogh, E. (2014). Dynamic time warping averaging of time series allows faster and more accurate classification, *Proceedings of the 14th IEEE International Conference on Data Mining, Shenzhen, China*, pp. 470–479.

Petitjean, F. and Gançarski, P. (2012). Summarizing a set of time series by averaging: From Steiner sequence to compact multiple alignment, *Journal of Theoretical Computer Science* **414**(1): 76–91.

Petitjean, F., Ketterlin, A. and Gançarski, P. (2011). A global averaging method for dynamic time warping, with applications to clustering, *Pattern Recognition* **44**(3): 678–693.

Rabiner, L.R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition, *Proceedings of the IEEE* **77**(2): 257–286.

Saito, N. (1994). *Local Feature Extraction and Its Applications Using a Library of Bases*, PhD thesis, Yale University, New Haven, CT.

Sakoe, H. and Chiba, S. (1971). A dynamic programming approach to continuous speech recognition, *Proceedings of the 7th International Congress of Acoustic, Budapest, Hungary*, pp. 65–68.

Soheily-Khah, S., Douzal-Chouakria, A. and Gaussier, E. (2016). Generalized k-means-based clustering for temporal data under weighted and kernel time warp, *Pattern Recognition Letters* **75**: 63–69.

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

Wang, L. and Jiang, T. (1994). On the complexity of multiple sequence alignment, *Journal of Computational Biology* **1**(4): 337–348.

Zhou, F. and De la Torre, F. (2009). Canonical time warping for alignment of human behavior, *in* Y. Bengio *et al.* (Eds.), *Advances in Neural Information Processing Systems 22*, Curran Associates, Inc., Vancouver, pp. 2286–2294.

Zhou, F. and De la Torre, F. (2016). Generalized canonical time warping, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **38**(2): 279–294.

Pierre-Francois Marteau received his MEng degree in computer science from the Graduate School of Engineering in Electronics, Computer Science and Telecommunications of Bordeaux (ENSEIRB) in 1984, and his PhD degree in computer science in 1988 from the Grenoble Institute of Technology (Grenoble INP). He experienced a post doctorate position at the University of Geneva in 1989, and at the Institute for Non Linear Sciences (INLS) at the University of California in San Diego in 1990. He then worked for eight years as an IT consultant in Bertin Technologies, a private company in Paris, before joining in 1999 the Computer Science Lab at the University of Southern Brittany, where he is now a professor. He joined the Institute for Research in Computer Science and Stochastic Systems (IRISA) in 2012. His current research interests include pattern recognition and machine learning with applications in sequential (symbolic and digital) data processing and security.

Received: 8 July 2018
Revised: 16 September 2018
Re-revised: 1 October 2018
Accepted: 23 October 2018