

# Covariance Features for Trajectory Analysis

Talha Karadeniz, Hakan Hadi Maras

*Department of Computer Engineering, Cankaya University,  
Ankara, Turkey  
hhmaras@cankaya.edu.tr*

**Abstract**—In this work, it is demonstrated that covariance estimator methods can be used for trajectory classification. It is shown that, features obtained via shrunk covariance estimation are suitable for describing trajectories. Compared to Dynamic Time Warping, application of explained technique is faster and yields more accurate results. An improvement of Dynamic Time Warping based on counting statistical comparison of base distance measures is also achieved. Results on Australian Sign Language and Character Trajectories datasets are reported. Experiment realizations imply feasibility through covariance attributes on time series.

**Index Terms**—Covariance matrices; Data mining; Sign language; Time series analysis.

## I. INTRODUCTION

In this work, focus is directed on the usage of covariance matrices for trajectory categorization. It is possible to establish an efficient analysis on high dimensional spaces via shrunk covariance matrices [1]. To our knowledge, no shrunk covariance technique is applied to time series, especially in the categorization task. So, it is decided to test the efficiency of these optimization-based covariance derivations on two datasets and compare the results with Dynamic Time Warping [DTW], a classical algorithm.

Since DTW is an elegant matching routine, application of almost all relatively new distance measures such as Least Common Subsequence [2] and Edit Distance on Real Sequence [3] yield very close results [4]. So, despite the lack of a guaranteed experimenting conclusion, relying on an extensive analysis [4], a comparison between developed algorithm and the classical DTW is considered sufficient. From Hidden Markov Models [5] and ‘elbow’ reductions [6] to kernel-based representations [7], there are various ways of handling trajectory analysis tasks but none of these are covariance-based except [8], where covariance matrices are instrumented to develop DTW rather than extracting explicit vector set descriptions. So, in terms of feature formation, that is, deduction of fixed-dimensional vectors from ordered vector sets, literature is still lacking in covariance-based analysis; especially when it comes to shrunk models.

Nowadays, trajectory analysis is more important. For we have tasks of manoeuvre [9] and driver intention [10] forecasting. Such kind of security applications need fast

spatial time-series categorization. Our work was started for the sake of medical operation improvement. A module of gesture categorization is built after the engineering of the method.

This report is organized as follows: in the second section, we give the steps of Dynamic Time Warping, Improved Dynamic Time Warping, newly developed feature extraction algorithm and the comparison tool Spectral Clustering. In the second section, accuracy and speed results are summarized. Finally, in the third section, an analysis with potential future work route suggestion is demonstrated.

## II. ALGORITHM

Let  $U = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_M\}$  and  $V = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N\}$  be two ordered vector sets where each  $\mathbf{u}_i, \mathbf{v}_i \in \mathbb{R}^D$ .

### A. Dynamic Time Warping

By Dynamic Time Warping, one can have an optimal match between vector sets, minimizing the sum of Euclidean distances under some special constraints such as boundary, continuity and monotonicity. Suppose that  $W = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{|W|}\}$  is a matching. Then the following conditions must be satisfied:

1. Boundary:  $\mathbf{w}_{|W|} = (M, N)$  and  $\mathbf{w}_1 = (1, 1)$ .
2. Continuity: if  $\mathbf{w}_i = (w_{i1}, w_{i2})$  and  $\mathbf{w}_{i+1} = (w_{(i+1)1}, w_{(i+1)2})$ , then  $\mathbf{w}_{(i+1)j} - \mathbf{w}_{ij} \leq 1$ .
3. Monotonicity: if  $\mathbf{w}_i = (w_{i1}, w_{i2})$  and  $\mathbf{w}_{i+1} = (w_{(i+1)1}, w_{(i+1)2})$ , then  $\mathbf{w}_{(i+1)j} - \mathbf{w}_{ij} \geq 0$ .

The solution is usually obtained by dynamic programming, with a complexity of  $O(MN)$ .

### B. Improved Dynamic Time Warping

Let  $\bar{d}$  be the mean of all Euclidean distances of input dataset. Let  $W = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{|W|}\}$  be the matching obtained between  $U$  and  $V$ . Assume that

$$f_1(\mathbf{w}_i) = \begin{cases} 1 & \text{if } \|\mathbf{u}_{\mathbf{w}_{i1}} - \mathbf{v}_{\mathbf{w}_{i2}}\| < \bar{d} \\ 0 & \text{otherwise} \end{cases}. \quad (1)$$

Define

$$s_{UV,1} = \left( \sum_{i=1}^{|W|} f_1(\mathbf{w}_{ij}) \right) / |W|. \quad (2)$$

Manuscript received 16 July, 2017; accepted 5 February, 2018.

This exploration is conducted for the Surgical Navigation Project (CAN) which is supported by TUBITAK (113S094). The engineering team would like to thank TUBITAK support for realizing this study.

Multiplying this scalar with the usual DTW distance yields a slightly improved result. Key idea here is to induce a scaling factor via the exploration of inter-vector distances and inter-set warping matches. Any such factor may contribute to overall accuracy.

Now, let  $\bar{d}_{local}$  be the mean of Euclidean distances of all tuples  $\mathbf{u} \in U$  and  $\mathbf{v} \in V$ . Suppose that

$$f_2(\mathbf{w}_i) = \begin{cases} 1 & \text{if } \|\mathbf{u}_{w_{i1}} - \mathbf{v}_{w_{i2}}\| < \bar{d}_{local} \\ 0 & \text{otherwise} \end{cases}. \quad (3)$$

Similar to (2), define

$$s_{UV,2} = \left( \sum_{i=1}^{|W|} f_2(\mathbf{w}_{ij}) \right) / |W|. \quad (4)$$

Now one can use  $s_{UV,1}$  and  $s_{UV,2}$  to fine-tune DTW distance  $D(U,V)$  as  $s_{UV,k} \times D(U,V)$  for  $k=1,2$  or in form  $s_{UV,1} \times s_{UV,2} \times D(U,V)$  to fuse these two measures.  $(s_{UV,1} + s_{UV,2}) \times D(U,V)$  is also observed to be efficient.

### C. Covariance Features for Trajectory Analysis

Covariance Features for Trajectories (CFT) is constructed by adapting Ledoit-Wolf (LW) [1] and Oracle Approximating Shrinkage (OAS) [11] to time-series. Common computation method shared in the application of these two techniques is that one have a destination estimate  $\hat{\mathbf{F}}$  and an initial matrix  $\hat{\mathbf{S}}$ , which are usually scaled identity matrix and empirical covariance, respectively. Optimization task

$$\begin{aligned} \min_{\lambda} \quad & E[\hat{\Sigma} - \Sigma \frac{2}{F}], \\ \text{such that} \quad & \hat{\Sigma} = \lambda \hat{\mathbf{F}} + (1-\lambda)\hat{\mathbf{S}}, \end{aligned} \quad (5)$$

is solved in order to find  $\lambda$ .

Let

$$\boldsymbol{\mu}_U = \sum_{i=1}^M \mathbf{u}_i, \quad (6)$$

be the mean vector of  $U$ ,  $\mathbf{I}$  be the identity matrix and empirical covariance of  $U$  be

$$\mathbf{S}_U = \sum_{i=1}^M (\mathbf{u}_i - \boldsymbol{\mu}_U)(\mathbf{u}_i - \boldsymbol{\mu}_U)^T. \quad (7)$$

Additionally, if  $\hat{\nu} = \text{tr}(\hat{\mathbf{S}}_U) / D$ , then  $\hat{\mathbf{F}}$  and  $\hat{\mathbf{S}}$  are set as  $\hat{\mathbf{F}} = \hat{\nu}\mathbf{I}$  and  $\hat{\mathbf{S}} = \mathbf{S}_U$ , respectively.  $\lambda$  can be found with different algorithms.

#### 1) LW Estimation

$\lambda$  is chosen as [1]

$$\lambda = \frac{\sum_{i=1}^M \mathbf{u}_i \mathbf{u}_i^T - \hat{\mathbf{S}} \frac{2}{F}}{M^2(\text{tr}(\hat{\mathbf{S}}^2) - \text{tr}^2(\hat{\mathbf{S}}) / D)}. \quad (8)$$

#### 2) OAS Estimation

Relying on the iteration steps [5]:

$$\begin{cases} \hat{\Sigma}_j = \lambda_j \hat{\mathbf{F}} + (1-\lambda_j)\hat{\mathbf{S}}, \\ \lambda_{j+1} = \frac{(1-2/D)\text{tr}(\Sigma_j \hat{\mathbf{S}}^2) + \text{tr}^2(\hat{\Sigma}_j)}{(M+1-2/D)\text{tr}(\Sigma_j \hat{\mathbf{S}}^2) + (1-M/D)\text{tr}^2(\hat{\Sigma}_j)}. \end{cases} \quad (9)$$

OAS limit is calculated as

$$\lambda = \frac{(1-2/D)\text{tr}(\hat{\mathbf{S}}^2) + \text{tr}^2(\hat{\mathbf{S}})}{(M+1-2/D)(\text{tr}(\hat{\mathbf{S}}^2) - \text{tr}^2(\hat{\mathbf{S}}) / D)}. \quad (10)$$

Now, given LW (8) and OAS (10) shrinkage models, it is appropriate to formulate an Ordered Representation Mapping [ORM], where, given its index  $i$  in the corresponding set  $U$ , a vector  $\mathbf{u}_i = (u_{i1}, u_{i2}, \dots, u_{iD})$  is mapped as

$$\phi_{b,m,n}(\mathbf{u}_i) = (b^{-i/m}, b^{-2i/m}, \dots, b^{-ni/m}, u_{i1}, u_{i2}, \dots, u_{iD}), \quad (11)$$

where  $b$ ,  $m$  and  $n$  are parameters of the function.

Suppose that  $\Sigma_{\phi_{b,n}(\mathbf{U})}$  is the LW covariance estimation of the image of set  $U$ . Similarly, let  $\Sigma_{\phi_{b,m,n}(\mathbf{U})}$  be the OAS estimation. Although these outputs are matrices, one can easily vectorise each by a row concatenation function  $\theta: \mathbb{R}^{(D+n) \times (D+n)} \rightarrow \mathbb{R}^{(D+n)^2}$  defined as

$$\begin{aligned} \theta \left( \begin{array}{ccc} a_{11} & \dots & a_{1(D+n)} \\ \vdots & \ddots & \vdots \\ a_{(D+n)1} & \dots & a_{(D+n)(D+n)} \end{array} \right) = \\ = (a_{11}, \dots, a_{1(D+n)}, \dots, a_{(D+n)1}, \dots, a_{(D+n)(D+n)}). \end{aligned} \quad (12)$$

In this case, CFT formulation is

$$\begin{aligned} (\theta(\Sigma_{\phi_{b,m,n}(\mathbf{U})}^{-1}) / |\theta(\Sigma_{\phi_{b,m,n}(\mathbf{U})}^{-1})|, \\ \oplus \\ (\theta(\Sigma_{\phi_{b,m,n}(\mathbf{U})}^{-1}) / |\theta(\Sigma_{\phi_{b,m,n}(\mathbf{U})}^{-1})|). \end{aligned} \quad (13)$$

That is, vectorised and normalized covariance matrix inverses are concatenated to build the final feature.

### D. Spectral Clustering

To follow a similar route noted in [2], Spectral Clustering [12] is selected, where an affinity matrix can be specified according to which the analysis can be done. Assume that we cluster  $N$  inputs into  $k$  clusters and have a similarity matrix  $\mathbf{A}^{N \times N}$ . Spectral Clustering is done as follows:

4. Let  $\mathbf{D}^{N \times N}$  be a diagonal matrix with elements

$$\mathbf{D}_{ii} = \sum_{j=1}^N \mathbf{A}_{ij}. \text{ Build the matrix } \mathbf{L} = \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}.$$

5. Take the  $k$  largest eigenvectors of  $\mathbf{L}$  to form the

matrix  $\mathbf{X}^{N \times k}$  where each column corresponds to an eigenvector.

6. Row-normalize  $\mathbf{X}$  and assign the result to  $\mathbf{Y}$ .

7. Apply k-means to the row vectors of  $\mathbf{Y}$ .

8. Assign the  $i$ -th input to cluster  $c$  if and only if the corresponding row of  $\mathbf{Y}$  is assigned to  $c$ .

### III. EXPERIMENTS

Experiments are conducted on two datasets retrieved from UCI Repository [13]. First, basic covariance features (which were induced from the initial form of (11), where we had no parameter at all) tested on Australian Sign Language (ASL) collection [14]. Second, Character Trajectories [15], which is a set of recorded handwriting character tips. Coding is done with Noweb [16] and scikit-learn [17].

Testing scheme summarized in [2] is repeated except that for clustering rather than hierarchical, spectral form is utilized since these are functionally similar with respect to the metric quality measurement.

#### A. ASL

10 words selected: Norway (Fig. 1), cold (Fig. 2), crazy (Fig. 3), eat, forget, happy, innocent, later, lose, spend. For each word, 5 trajectories are extracted. Each tuple of word vector sets, that is, a total of 10 vectors are clustered with respect to DTW, Improved DTW (2), (4) and CFT (13). If after clustering, each element of this set is labelled correctly, then the operation is counted as accurate. Number of such tests for a 10-word collection is 45. Accuracy and speed comparison is given in Table I and Table II, respectively.

CFT parameters (11) are chosen as  $b = e$ ,  $m = 1$  and  $n = 1$ . DTW-A is improved DTW with scaling factor (2).

TABLE I. ACCURACY ON ASL.

Algorithm	Number of Correct Clusterings (out of 45)	Accuracy
DTW	19	42.2 %
DTW-A	22	48.0 %
CFT	24	53.3 %

TABLE II. EXPERIMENT DURATION FOR ASL.

Algorithm	Duration
DTW	301.76 s.
DTW-A	368.07 s.
CFT	2.22 s.

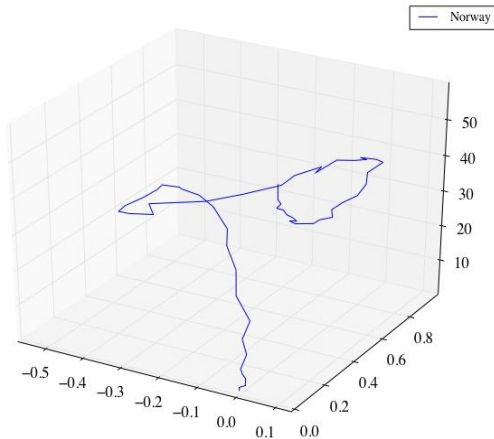


Fig. 1. 'Norway' trajectory. Values on horizontal axes stands for x and y coordinates. Vertical axis is the vertex index.

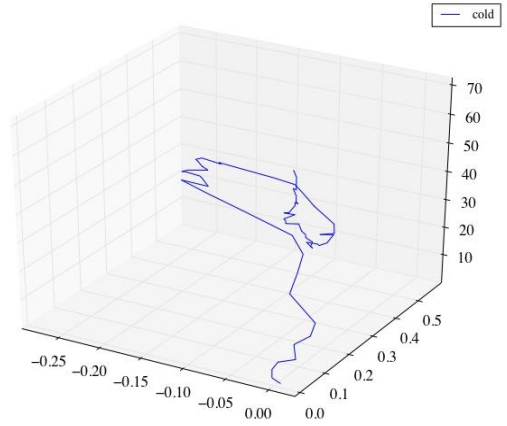


Fig. 2. 'Cold' trajectory. Notice the horizontal variation difference between this and Fig. 1, which is very explanatory on the benefit from CFT selection.

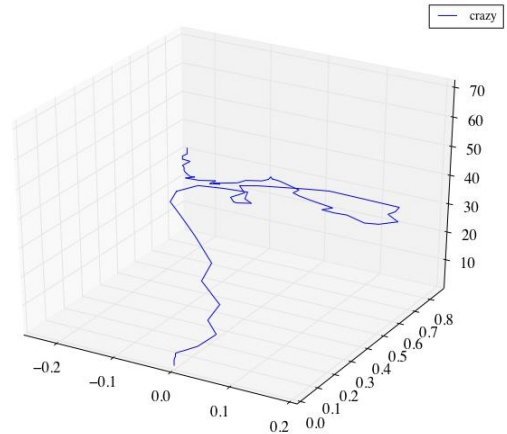


Fig. 3. 'Crazy' trajectory. Again, horizontal amplitude is separate from that of 'cold'. On the other hand, here, change of vertical component with respect to time is more uniform which is not this case in 'Norway' and 'cold'. Situation is handled via Ordered Representation Mapping (II-C): that is, index-to-feature mapping to measure correlation with time.

#### B. Character Trajectories

10 characters selected to test accuracy of DTW, modified DTW and CFT: a, b, c, d, e, g, h, l, m and n. First 5 trajectories of each are collected to complete the route given in [18]. Results are given in Table III and Table IV.

DTW-B is obtained by adding (4) to (2) and hence scaling DTW as  $(s_{UV,1} + s_{UV,2}) \times D(U, V)$ .

TABLE III. EXPERIMENT DURATION FOR CHARACTER TRAJECTORIES.

Algorithm	Duration
DTW	1301.7 s.
DTW-B	1538.2 s.
CFT	2.86 s.

CFT (11) is selected as  $b = 0.4 \times e$ ,  $m = 20$  and  $n = 3$ .

### IV. ANALYSIS

One advantage of choosing DTW over CFT is related to parameter issue since one needs to consolidate the background of CFT by exploring the nature of the exponential decay multitude noted in (11). Thus DTW can be applied to many cases more easily without any further grid search.

On the other hand, CFT is much more suitable for real-time classification than DTW since it is approximately 100

times faster. This speed advantage is more important in case where average length of trajectories is high.

Regarding the results noted in former section, one can remember the ‘peeking at the test data’ warning noted in [4] and question CFT. This is reasonable for we need to confirm superiority after cross-validation-based parameter selection and accuracy measurement but such incompleteness does not show that achieved accuracy rates are valueless.

From the experiments, one can deduce that, despite the large-scale or high-dimensional targeting intention behind Minimum Mean-Squared Error covariance estimates [1], [11], such a modelling scheme may also be adopted to 2-dimensional or 3-dimensional time-series tasks. Moreover, on noisy datasets such as ASL, benefit from CFT is more visible compared to those supplied on Character Trajectories, where, though not negligible, an increased clustering accuracy of 5 % is observed.

## V. CONCLUSIONS

Trajectory analysis is beneficial for many applications ranging from driver intention forecasting [9] to aircraft engineering [19]. In this work, a feature – built on shrunk covariance matrices – for trajectories is developed. Moreover, test results indicate that the proposed method is more accurate and at the same time more efficient than its competitors. Normally, covariance matrices are meant to be calculated on unordered sets; to overcome this handicap, exponential decay based Ordered Representation Mappings are introduced. That is, vectors are mapped to a higher dimensional space with respect to their order in the corresponding set. Tests on sign language and handwriting characters indicate that, linear with the length of the trajectory, constructed features can be utilized in real-time applications in order to get fast and accurate results. Additionally, in spite of the simplistic counting ratio methodology noted here, future study is open to new techniques of improving DTW through more elegant matching sequence analysis.

## REFERENCES

- [1] L. Olivier, M. Wolf, “A well-conditioned estimator for large-dimensional covariance matrices”, *Journal of multivariate analysis*, vol. 88, no. 2, pp. 365–411, 2004. DOI: 10.1016/S0047-259X(03)00096-4.
- [2] M. Vlachos, G. Kollios, D. Gunopulos, “Discovering similar multidimensional trajectories”, in *Proc. IEEE 18<sup>th</sup> Int. Conf. Data Engineering*, 2002, pp. 673–684. DOI: 10.1109/ICDE.2002.994784.
- [3] L. Chen, M. T. Ozsu, V. Oria, “Robust and fast similarity search for moving object trajectories”, in *Proc. Int. Conf. Management of Data (ACM SIGMOD 2005)*, 2005, pp. 491–502. DOI: 10.1145/1066157.1066213.
- [4] X. Wang *et al.*, “Experimental comparison of representation methods and distance measures for time series data”, *Data Mining and Knowledge Discovery*, vol. 26, no. 2, pp. 275–309, 2013. DOI: 10.1007/s10618-012-0250-5.
- [5] J. C. Nascimento, M. A. T. Figueiredo, J. S. Marques, “Trajectory classification using switched dynamical hidden Markov models”, *IEEE Trans. Image Processing*, vol. 19, no. 5, pp. 1338–1348, 2010. DOI: 10.1109/TIP.2009.2039664.
- [6] H. T. Pham, J. Kim, Y. Won, “A cost effective method for matching the 3d motion trajectories”, in *IT Convergence and Security 2012*, 2013, pp. 889–895. DOI: 10.1007/978-94-007-5860-5\_107.
- [7] W. Y. Lin, C. Y. Hsieh, “Kernel-based representation for 2D/3D motion trajectory retrieval and classification”, *Pattern Recognition*, vol. 46, no. 3, pp. 662–670, 2013. DOI: 10.1016/j.patcog.2012.09.014.
- [8] Z. Banko, J. Abonyi, “Correlation based dynamic time warping of multivariate time series”, *Expert Systems with Applications*, vol. 39, no. 17, pp. 12814–12823, 2012. DOI: 10.1016/j.eswa.2012.05.012.
- [9] E. Ward, J. Folkesson, “Multi-classification of driver intentions in yielding scenarios”, *IEEE 18<sup>th</sup> Int. Conf. Intelligent Transportation Systems (ITSC 2015)*, 2015, pp. 678–685. DOI: 10.1109/ITSC.2015.116.
- [10] L. Peng, *et al.*, “Classification of highway lane change behavior to detect dangerous cut-in maneuvers”, *The 95<sup>th</sup> Annual Meeting Transportation Research Board (TRB)*, vol. 2, 2016.
- [11] Y. Chen *et al.*, “Shrinkage algorithms for MMSE covariance estimation”, *IEEE Trans. Signal Processing*, vol. 58, no. 10, pp. 5016–5029, 2010. DOI: 10.1109/TSP.2010.2053029.
- [12] A. Y. Ng, M. I. Jordan, Y. Weiss, “On spectral clustering: Analysis and an algorithm”, *Advances in neural information processing systems 2*, pp. 849–856, 2002.
- [13] UC Irvine Machine Learning Repository. [Online]. Available: <http://archive.ics.uci.edu/ml>
- [14] Australian Sign Language signs Data Set. [Online]. Available: <http://archive.ics.uci.edu/ml/datasets/Australian+Sign+Language+Signs>
- [15] Character Trajectories Data Set. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/Character+Trajectories>
- [16] N. Ramsey, “Noweb—a simple, extensible tool for literate programming”, [Online]. Available: <https://www.cs.tufts.edu/~nr/pubs/lpsimp.pdf>
- [17] Machine Learning in Python, scikit-learn. [Online]. Available: <http://scikit-learn.org/stable/>
- [18] E. J. Keogh, M. J. Pazzani, “Scaling up dynamic time warping for datamining applications”, in *Proc. 6<sup>th</sup> Int. Conf. Knowledge Discovery and Data Mining (ACM SIGKDD 2000)*, 2000, pp. 285–289. DOI: 10.1145/347090.347153.
- [19] S. T. ul Islam Rizvi, H. Linshu, X. Dajun, “Optimal trajectory analysis of hypersonic boost-glide waverider with heat load constraint”, *Aircraft Engineering and Aerospace Technology: An International Journal*, vol. 87, no. 1, pp. 67–78, 2014. DOI: 10.1016/j.dt.2015.06.003.