



## MULTIDIMENSIONAL SEQUENCE CLUSTERING WITH ADAPTIVE ITERATIVE DYNAMIC TIME WARPING

Sergii Mashtalir <sup>1)</sup>, Olena Mikhnova <sup>2)</sup>, Mykhailo Stolbovyi <sup>1)</sup>

<sup>1)</sup> Informatics department, Kharkiv National University of Radio Electronic  
61166, Nauky Ave. 14, Kharkiv, Ukraine, sergii.mashtalir@nure.ua, st.mihail92@gmail.com

<sup>2)</sup> Cybernetics department, Kharkiv Petro Vasylenko National Technical University of Agriculture,  
61002, Alchevskyyh Str. 44, Kharkiv, Ukraine, mikhnova@yahoo.co.uk

### Paper history:

Received 25 October 2018

Received in revised form 30 December 2018

Accepted 28 January

Available online 31 March 2019

### Keywords:

segmentation;

clustering;

multidimensional sequence;

video processing;

adaptive iterative dynamic time warping;

Kohonen self-organizing map.

**Abstract:** Multimedia sequence matching is an urgent problem nowadays for the field of artificial intelligence. Despite great progress in this field of computer science, huge data arrays require near to real time processing, which significantly limits applicable methods. In this paper, the authors make an attempt of time series approach modification and enhancement with orientation for non-stationary data of different length. The processing procedure is enriched by sequences alignment with iterative dynamic time warping which implies matching two temporal data segments. Computational complexity is reduced due to Kohonen self-organizing maps applied for the purpose of clustering. Mathematical presentation is given in scalar, vector and matrix forms in order to cover all the possible use cases. An example of video sequence processing with the novel approach is provided to show its efficiency. The proposed technique can also be successfully implemented for natural language and signal processing, bioinformatics and financial data analysis.

Copyright © Research Institute for Intelligent Computer Systems, 2019.  
All rights reserved.

## 1. INTRODUCTION

With the rapid growth of data freely available in the Internet, matching multimedia content has recently become an urgent problem [1-4]. This article analyses prerequisites for content matching with limitations of existing techniques. The prime attention is given to video sequence processing as the most complicated type of multimedia. But before any artificial intelligence method is applied, video sequence needs to be segmented and classified according to some criterion.

Aside from video, the problem of temporal sequence segmentation and clustering exists in many practical applications [5-7], and a great number of algorithms have been developed by now to cope with this problem [8-13]. When the sequences under research are of different length, the situation becomes even more complicated and influences inability of applying traditional metrics for cluster analysis. In such cases, similarity measures and functions along with dimensionality reduction techniques are usually employed, among which are:

Minkowski distances (including Euclidean and Manhattan distances), maximum distance, Mahalanobis distance, Hamming distance, Gaussian affinities, Cosine similarities, Jaccard index, cross-correlation function, Landmarks similarity, Dynamic Time Warping (DTW), Longest Common Substring, spectral decomposition through Discrete Fourier Transform, Discrete Wavelet Transform, Singular Value Decomposition, Piecewise Aggregate / Constant / Linear Approximation, Chebyshev Polynomials, just to name a few [14-16].

The main challenges here are linked to multidimensionality and processing time. The former involves the curse of dimensionality and ambiguity of similarity measures for multidimensional feature space; and the latter should obviously tend to near real time. Dimensionality reduction without losing essential information is the goal of any approach designed to cope with high-dimensional time sequences. In this relation, DTW should be mentioned first of all [16, 17]. It enables evaluation of distance between any two sample series from a sequence of observations.

Thus, consider two one-dimensional sequences of different length  $X = \{x(1), x(2), \dots, x(k), \dots, x(N)\}$  and  $Y = \{y(1), y(2), \dots, y(k), \dots, y(M)\}$ ,  $N \neq M$ , the distance between which is to be estimated. This distance should then be suitable for clustering with any of the known approaches. To do so, the distance matrix ( $N \times M$ ) is considered in an arbitrary metric for all the sample elements  $d(x(k), y(l))$   $k = 1, 2, \dots, N; l = 1, 2, \dots, M$ . So called 'warping path' is computed using this matrix. It is formed by the sequence  $W = \{w_1, w_2, \dots, w_L\}$ ,  $w_q = d(x(k), y(l))_q$ ,  $\max(N, M) \leq L < M + N + 1, q = 1, 2, \dots, L$ , and it is a basis for similarity estimation between series  $X$  and  $Y$ . Warping path serves as a distance between the series, providing minimum to the following criterion [18]:

$$DTW(x, y) = \min_W \left\{ \frac{1}{L} \sum_{q=1}^L d(x(k), y(l))_q \right\}. \quad (1)$$

Minimization of criterion (1) is performed with the procedure of dynamic programming based on recurrent relation  $D(k, l) = d(x(k), y(l)) + \min\{D(k, l-1), D(k-1, l), D(k-1, l-1)\}$ , where  $D(k, l)$  is a cumulative distance between sequence elements  $x(k), y(l)$ . With this,  $DTW(x, y) = D(N, M)$ . This approach was expanded for multidimensional sequences  $x(k), y(l) \in R^N$  in [19]. Euclidean metric, here, is used as a distance between observations

$$d^2(x(k), y(l)) = \|x(k) - y(l)\|^2. \quad (2)$$

In case of matrix signal processing, e.g., video sequence processing, when  $x(k) = \{x_{i_1, i_2}(k)\} \in R^{n \times v}$ ,  $y(l) = \{y_{i_1, i_2}(l)\} \in R^{n \times v}$ , Frobenius norm can be easily introduced

$$d(x(k), y(l)) = (Sp(x(k) - y(l))(x(k) - y(l))^{1/2}), \quad (3)$$

that generalizes the metric (2).

The main challenges with DTW usage appear when long time series are to be analyzed, such as video sequences. Dynamic programming (and the 'curse of dimensionality' connected to it) influences tremendous computational complexity in this case,

which makes this approach ineffective [16]. With this in mind, modification of DTW approach is reasonable for processing of time series with sufficient length.

## 2. TIME SERIES MODEL DEVELOPMENT BASED ON ITERATIVE DYNAMIC TIME WARPING

Computational complexity can be significantly shrunk due to iterative dynamic time warping approach discussed in [20, 21]. It implies reduction of the analyzed series with the help of some estimates given to particular sequence intervals. With this in mind, if two series  $X$  and  $Y$  are defined with  $N$  and  $M$  observations respectively, five points are selected from each series:  $x(1), x(\lfloor \frac{N}{4} \rfloor), x(\lfloor \frac{N}{2} \rfloor), x(\lfloor \frac{3N}{4} \rfloor), x(N)$  and  $y(1), y(\lfloor \frac{M}{4} \rfloor), y(\lfloor \frac{M}{2} \rfloor), y(\lfloor \frac{3M}{4} \rfloor), x(M)$  (where  $\lfloor z \rfloor$  is the quotient of  $z$ ). After that, mean values  $\bar{x}(1), \bar{x}(2), \bar{x}(3), \bar{x}(4), \bar{y}(1), \bar{y}(2), \bar{y}(3), \bar{y}(4)$  are computed on intervals  $x(1) - x(\lfloor \frac{N}{4} \rfloor), x(\lfloor \frac{N}{4} \rfloor + 1) - x(\lfloor \frac{N}{2} \rfloor), x(\lfloor \frac{N}{2} \rfloor + 1) - x(\lfloor \frac{3N}{4} \rfloor), x(\lfloor \frac{3N}{4} \rfloor + 1) - x(N)$  and  $y(1) - y(\lfloor \frac{M}{4} \rfloor), y(\lfloor \frac{M}{4} \rfloor + 1) - y(\lfloor \frac{M}{2} \rfloor), y(\lfloor \frac{M}{2} \rfloor + 1) - y(\lfloor \frac{3M}{4} \rfloor), y(\lfloor \frac{3M}{4} \rfloor + 1) - x(M)$ , which form reduced sequences. These observations produce  $(4 \times 4)$ -matrix of distances, the following cumulative distance is computed on its basis:

$$DTW(x, y) = DTW(4, 4). \quad (4)$$

If the estimates obtained from (4) are not satisfactory from the point of precision, then each interval is divided into two based on the reduced series. Thus, the mean values  $\bar{x}(1), \bar{x}(2), \bar{x}(3), \dots, \bar{x}(8), \bar{y}(1), \bar{y}(2), \bar{y}(3), \dots, \bar{y}(8)$  are obtained and  $DTW(8, 8)$  is computed. Series reduction process repeats until needed result is obtained, and the number of intervals grows in geometric progression. From the computational point, it is more convenient to increase the interval number in arithmetic progression, e.g.,  $g = 4, 5, 6, \dots, R \leq \min\{N, M\}$ . Thus, each series corresponds to the sequence with  $g+1$  scores. Turning back to clustering problem of the time sequence  $x_1, x_2, \dots, x_q, \dots, x_Q$ , each of which has  $N_1, N_2, \dots, N_q, \dots, N_Q$  observations respectively, it can be converted to the problem of sequence

processing  $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_q, \dots, \bar{x}_Q$ , where each sequence has  $g + 1$  scores.

The aforementioned approach is only applicable for situation when series properties remain unchanged at each of the formed intervals. The latter condition is not always provided for real video processing applications. That is why mean based clustering may turn ineffective. From the point of general time series clustering problem, the technique discussed above relates to feature based approaches. Approaches based on time series models seem to be more promising for the problems of non-stationary sequence processing [8]. It should be mentioned here that the latter approaches are developed for the sequences of the same length. Therefore, it makes sense to merge time series model based approach with iterative dynamic time warping.

Thus far, consider a set of temporal sequences with different length, each of which is divided into  $p$  intervals, and these intervals may contain different number of observations in each series  $x_q$ . Let us denote an observation of  $q$ -th series in  $r$ -th interval as  $x_{qr}(\tau)$ , where  $\tau$  is the number of this observation in the interval, and this number changes from 1 to maximum length of this interval. Assume that each observation corresponds with the following linear adaptive model:

$$x_{qr}(\tau) = a_{qr} + b_{qr} \tau = A_{qr}^T \bar{\tau}, \quad (5)$$

where  $A_{qr}^T = (a_{qr}, b_{qr})^T$ ,  $\bar{\tau} = (1, \tau)^T$ .

Model parameter setting is performed with the recursive least squares method:

$$\begin{cases} A_{qr}^T(\tau) = A_{qr}^T(\tau-1) + P_{qr}(\tau)((x_{qr}(\tau) - A_{qr}^T(\tau-1)\bar{\tau})\bar{\tau}), \\ P_{qr}(\tau) = P_{qr}(\tau-1) - \frac{P_{qr}(\tau-1)\bar{\tau}\bar{\tau}^T P_{qr}(\tau-1)}{1 + \bar{\tau}^T P_{qr}(\tau-1)\bar{\tau}}. \end{cases} \quad (6)$$

As a result of such data processing, each  $q$ -th series  $x_q$  corresponds with  $(2g \times 1)$  vector  $A_q$ , clustering is further performed on its basis. It should also be noted that series stationarity at partition regions influences  $b_{qr}$  parameters tend to zero, i.e. turning back to iterative dynamic time warping. In case of multidimensional time vectors  $x_q \in R^n$ , the following model corresponds with each series:

$$\bar{x}_{qr}(\tau) = A_{qr}^T \bar{\tau}, \quad (7)$$

where  $A_{qr}$  is  $(n \times 2)$ -matrix of parameters to be estimated.

To configure model parameters (7), recursive algorithm of the following type can be used:

$$\begin{cases} A_{qr}^T(\tau) = A_{qr}^T(\tau-1) + ((x_{qr}(\tau) - A_{qr}^T(\tau-1)\bar{\tau})\bar{\tau})P_{qr}(\tau), \\ P_{qr}(\tau) = P_{qr}(\tau-1) - \frac{P_{qr}(\tau-1)\bar{\tau}\bar{\tau}^T P_{qr}(\tau-1)}{1 + \bar{\tau}^T P_{qr}(\tau-1)\bar{\tau}}. \end{cases} \quad (8)$$

Thus, each series  $x_q$  corresponds with  $r$  matrices that can be merged into a complex  $(n \times 2g)$ -matrix  $A_q$ . It should be mentioned here that the relations (7) and (8) have been used for online video frame segmentation.

Finally, when we are dealing with  $x_{qr}(\tau) = \{x_{qr i_2}(\tau)\} \in R^{n \times v}$  problems, the following mathematical model can correspond with each sequence:

$$\hat{x}_{qr}(\tau) = A_{qr}^T \bar{\tau} B_{qr}, \quad (9)$$

where  $A_{qr}$  is  $(n \times 2)$ -matrix and  $B_{qr}$  is  $(1 \times v)$ -vector of parameters respectively.

The model parameters (9) can be configured with the modified algorithm. To do so, the following model can correspond with the equation (9):

$$\hat{x}_{qr}(\tau) = A_{qr}^T \bar{\tau} B_{qr}^B, \quad (10)$$

where  $\bar{\tau} B_{qr}^B = \bar{\tau} B_{qr}$  is  $(2 \times v)$ -matrix.

Least squares estimation can be written as follows for matrix  $A_{qr}$  in (10):

$$A_{qr} = p_{A_{qr}} R_{A_{qr}}^{-1},$$

where

$$\begin{cases} p_{A_{qr}} = \sum_{\tau} x_{qr}(\tau) \bar{\tau}_{qr}^{BT}, \\ R_{A_{qr}} = \sum_{\tau} \bar{\tau}_{qr}^B \bar{\tau}_{qr}^{BT}. \end{cases}$$

It can also be written in a recursive form:

$$\begin{cases} p_{A_{qr}}(\tau) = p_{A_{qr}}(\tau-1) + x_{qr}(\tau) \bar{\tau}_{qr}^{BT}(\tau), \\ R_{A_{qr}}(\tau) = R_{A_{qr}}(\tau-1) + \bar{\tau}_{qr}^B(\tau) \bar{\tau}_{qr}^{BT}(\tau). \end{cases} \quad (11)$$

By analogy to the previous manipulations, let us introduce the following model:

$$\hat{x}_{qr}(\tau) = \bar{\tau}_{qr}^A B_{qr}, \quad (12)$$

where  $\bar{\tau}_{qr}^A$  is  $(n \times 1)$ -vector.

Least squares estimation for the row-vector  $B_{qr}$  in (10) can be written as follows:

$$B_{qr} = R_{B_{qr}}^{-1} p_{B_{qr}},$$

where

$$\begin{cases} R_{B_{qr}} = \sum_{\tau} \bar{\tau}_{qr}^{AT} \bar{\tau}_{qr}^A, \\ p_{B_{qr}} = \sum_{\tau} x_{qr}^T(\tau) \bar{\tau}_{qr}^{AT}. \end{cases}$$

It can also be written in a recursive form:

$$\begin{cases} p_{A_{qr}}(\tau) = p_{B_{qr}}(\tau-1) + x_{qr}^T(\tau) \bar{\tau}_{qr}^{AT}(\tau), \\ R_{A_{qr}}(\tau) = R_{B_{qr}}(\tau-1) + \bar{\tau}_{qr}^{AT}(\tau) \bar{\tau}_{qr}^A(\tau). \end{cases} \quad (13)$$

As a result, the series  $x_q$  corresponds with  $(n \times 2g)$ -matrix  $A_q$  and  $(vg \times 1)$ -vector  $B_q^T$ , which are further used for clustering. Hence, the object for one-dimensional series clustering is  $(2g \times 1)$ -vectors  $A_q$ , and the object for multidimensional series clustering is  $(n \times 2g)$ -matrices  $A_q$ , and finally, the object for matrix series clustering is  $(n \times 2g)$ -matrix  $A_q$  and  $(vg \times 1)$ -vector  $B_q^T$ .

### 3. TIME SERIES CLUSTERING WITH KOHONEN SELF-ORGANIZING MAPS

In order to cluster the arrays of vectors  $A_q, B_q^T$  (for one-dimensional and multidimensional sequences) and matrices  $A_q$  (for multidimensional and matrix signals), we have used an approach based on centroid prototypes which implies partitioning of all the data array into  $m$  clusters with  $c_j$  centroids,  $j = 1, 2, \dots, m$ , needed to be set. Further, without losing generality, vectors needed for clustering will be denoted as  $B_q$  and matrices will be denoted as  $A_q$ .

Clustering problem has been solved with such a type of neural networks as Kohonen self-organizing maps [22]. First, address an issue of clustering (self-organizing) for vector signals  $B_q, q = 1, 2, \dots, Q$ . Self-organizing procedure implies two consequential

steps: competition and synapse adaptation. The procedure begins with rather arbitrary setting of centroids  $c_j^B(0)$ , which are then improved during adaptation. Processing starts from the first vector  $B_1$  and all the centroids  $c_j^B(0)$  in Euclidean metric:

$$d^2(B_1, c_j^B(0)) = \|B_1 - c_j^B(0)\|^2.$$

Then, the ‘winning’ centroid  $c_j^{B*}$  is selected, which is nearest to the vector  $B_1$  in the sense of distance (2), such that:

$$d(B_1, c_j^{B*}(0)) = \min_j d(B_1, c_j^B(0)). \quad (14)$$

This is the aim of competition, to set a winner  $c_j^{B*}(q-1)$  per each  $B_q$ .

Adaptation step assumes arrangement of the winning centroid location with the help of the following recursive relation (learning rule ‘the winner takes all’):

$$c_j^B(1) = \begin{cases} c_j^{B*}(0) + \eta(1)(B_1 - c_j^{B*}(0)) & \text{if } c_j^{B*}(0) \text{ is the winner,} \\ c_j^B(0) & \text{for all the rest centroids.} \end{cases}$$

Here,  $\eta(1)$  is a setting interval parameter that monotonically decreases during adaptation process, and it is usually specified according to empirical reasons. After  $q$ -th vector  $B_q$  has been transferred for processing and the winner  $c_j^{B*}(q-1)$  has been defined, adaptation process is implemented according to the relation:

$$c_j^B(q) = \begin{cases} c_j^{B*}(q-1) + \eta(q)(B_q - c_j^{B*}(q-1)) & \text{if } c_j^{B*}(q-1) \text{ is the winner,} \\ c_j^B(q-1) & \text{for all the rest centroids.} \end{cases} \quad (15)$$

The arrangement of location is performed until the whole sample collection  $B_1, B_2, \dots, B_Q$  exists. If centroids have not been stabilized until then with the following inequality fulfillment

$$\sum_{j=1}^m \|c_j^B(Q) - c_j^B(Q-1)\| \leq \varepsilon, \quad (16)$$

then  $B_1, B_2, \dots$  are transferred for processing again after  $B_Q$ . This process is repeated until the condition

(16) is held. Clustering result involves obtaining  $m$  stable values of centroids  $c_j^B$ . It should be noted that this result agrees with wide-spread  $k$ -means clustering, it only differs in processing time and requires much less computational resources [22, 23].

To solve matrix clustering problem  $A_1, A_2, \dots, A_Q$ , matrix modification of a self-organizing map can be used. With this in mind, the competition step is implemented with the help of the metric (3)

$$d(A_q, c_j^A(q-1)) = (Sp(A_q - c_j^A(q-1))(A_q - c_j^A(q-1)))^{1/2},$$

$$d(A_q, c_j^{A^*}(q-1)) = \min_j d(A_q, c_j^A(q-1)),$$

and the adaptation step is implemented with the help of the following recursive relation:

$$c_j^A(q) = \begin{cases} c_j^{A^*}(q-1) + \eta(q)(A_q - c_j^{A^*}(q-1)) & \text{if } c_j^{A^*}(q-1) \text{ is the winner,} \\ c_j^A(q-1) & \text{for all the rest centroids.} \end{cases} \quad (17)$$

it is easy to see that adaptation procedures (15) and

(17) are structurally agreed, which enables using the relation (17) even for vector sequences, especially as Frobenius norm is a generalization of Euclidean one.

#### 4. EXPERIMENTAL RESULTS OF VIDEO SEQUENCE SEGMENTATION

In order to analyze the efficiency of the proposed multidimensional time series clustering procedure, series of video sequences have been used as typical representatives for such kind of data. Fig. 1 shows an example of video sequence taken from ‘Destroyed in Seconds’ documentary cycle on the Discovery Channel. This sequence consists of several segments (500 frames total), illustrating a plane take-off with further crash-down that influenced combustion and explosion of an aircraft. A human expert can easily partition this video into 3 semantically meaningful consequential segments: runway acceleration phase shot from a single camera angle (frames from 1 to 90), take-off phase shot from another angle (frames from 91 to 266), crash-down phase shot from the third angle (frames from 277 to 500). Scene change is indicated in Fig. 1 with transition frames depicting the accident.

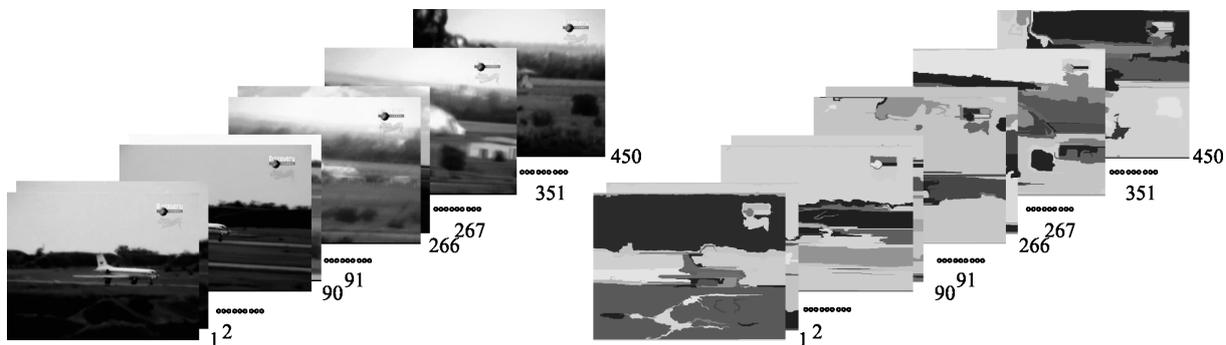


Figure 1 – Spatial and temporal segmentation of video frames

Temporal video segmentation (partitioning into scenes) with the proposed approach is given by the plot provided in Fig. 2. It can be clearly seen that the events of acceleration and take-off are distinctly identified by value peaks in the plot at corresponding moments of time. Thus, we may conclude that the proposed approach provides possibilities to perform video sequence clustering. In connection with this, it should be noted that transition between the first two segments are easily identified despite significant data changes are absent here, and the rest segment frames differ insignificantly. At the same time, according to the plot, the third segment is not that homogeneous as

the first two. It is mostly due to the fact that the plane crash-down only took place at the beginning, after a while it turned into a serious fire hazard as shown in frame 450 (see fig. 1). According to this, value peaks presented in the interval from 267 frame to 500 frame can be categorized as noise outliers, otherwise this segment of fire hazard may be partitioned into some more sub-segments indicating inflammation, small fire and serious hazard respectively (depending on the problem statement). Though, additional analysis of the latter segment using the proposed approach may help identifying more precise segment boundaries.

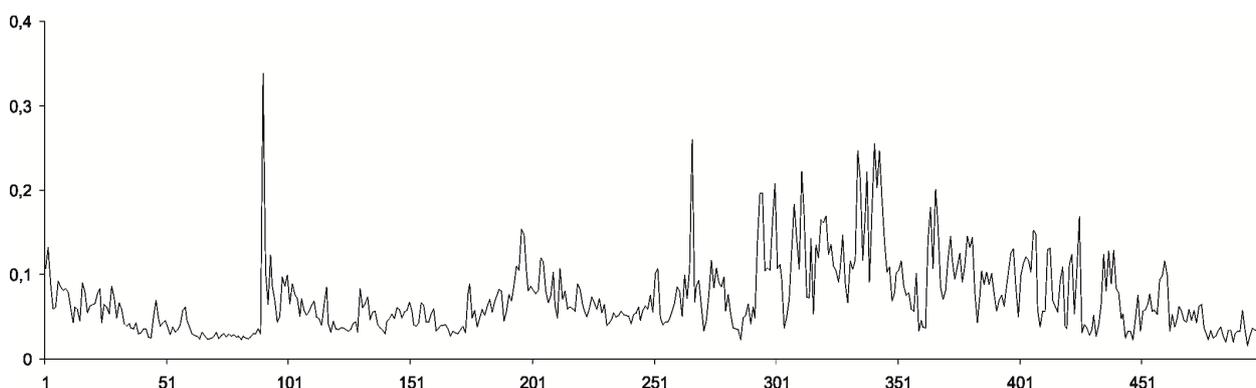


Figure 2 – Plot of temporal segmentation for the initial video sequence

## 5. CONCLUSION

Computationally simple procedure for time series clustering has been proposed in this paper (in scalar, vector and matrix forms). It grounds on the use of both modified iterative dynamic time warping and Kohonen self-organizing clustering map. In contrast to the early works [12, 13], the use of both proposed approaches allowed segmentation process in matrix form (more logical in terms of multidimensional data processing) with Frobenius norm, which made it possible to avoid the vectorization/devectorization process to increase the speed, which is especially important in video processing. It should also be noted that the experiments were conducted on more than 20 different videos from ‘Destroyed in Seconds’ documentary cycle and as a result of clustering, shots with content-based similar characteristics were obtained. The given here approach provides means for huge arrays of highly dimensional data processing within acceptable time. The novel approach was designed for video sequence processing, though it can also be successfully used in natural language and signal processing, DNA/RNA and financial data analysis just to name a few.

## 6. REFERENCES

- [1] H. Chen, D. Chen, S. Lee, “Object based video similarity retrieval and its application to detecting anchorperson shots in news video,” *Proceedings of the IEEE Fifth International Symposium on Multimedia Software Engineering*, Taichung, Taiwan, December 10-12, 2003, pp. 172-179.
- [2] M. Tang, S. Pongpaichet, R. Jain, “Research challenges in developing multimedia systems for managing emergency situations,” *Proceedings of the 2016 ACM Conference on Multimedia*, Amsterdam, the Netherlands, October 15-19, 2016, pp. 938-947.
- [3] D. Reynolds, R.A. Messner, “Video copy detection utilizing the log-polar transformation,” *International Journal of Computing*, vol. 15, issue 1, pp. 8-13, 2016.
- [4] N.I. Korsunov, D.A. Toropchin, “The method of finding the spam images based on the hash of the key points of the image,” *International Journal of Computing*, vol. 15, issue 4, pp. 259-264, 2016.
- [5] A. Valente et. al. “When does picture naming take longer than word reading?,” in: S. Sulpizio, S. Kinoshita (Eds.), *Bridging Reading Aloud and Speech Production*, vol. 7, article 31, Frontiers, Lausanne, 2016, pp. 83-93.
- [6] D. Jurafsky, J.H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 2-nd edition, Prentice Hall, New Jersey, 2008, 1032 p.
- [7] T. Tsai and S. Lee, “SimSearcher: A local similarity search engine for biological sequence databases,” *Proceedings of the IEEE Fifth International Symposium on Multimedia Software Engineering*, Taichung, Taiwan, December 10-12, 2003, pp. 305-312.
- [8] T.W. Liao, “Clustering of time series data,” *Pattern Recognition*, vol. 38, issue 11, pp. 1857-1874, 2005.
- [9] E.I. Keogh, S. Chu, D. Hart, M. Pazzani, “Segmenting time series: A survey and novel approach,” in: M. Last, A. Kandel, H. Bunke (Eds.), *Data mining in time series databases*, World Scientific Publ. Company, New Jersey, 2004, pp. 1-22.
- [10] C.C. Aggarwal, *Data Mining: The Textbook*, Springer, New York, 2015, 734 p.
- [11] C.C. Aggarwal, C.K. Reddy, *Data Clustering: Algorithms and Applications*, CRC Press, Boca Raton, 2014, 652 p.
- [12] S. Mashtalir, O. Mikhnova, “Detecting significant changes in image sequences,” in:

- A.E. Hassanien et. al. (Eds.), *Multimedia Forensics and Security*, Springer, Basel, 2017, pp. 161-191.
- [13] Zh. Hu, S.V. Mashtalir, O.K. Tyshchenko, M.I. Stolbovyi, "Clustering matrix sequences based on the iterative dynamic time deformation procedure", *International Journal of Intelligent Systems and Applications*, vol. 10, no. 7, pp. 66-73, 2018.
- [14] J. Han, M. Kamber, J. Pei, *Data Mining: Concepts and Techniques*, 3-rd edition, Elsevier, Amsterdam, 2013, 703 p.
- [15] F. Iglesias, W. Kastner, "Analysis of similarity measures in times series clustering for the discovery of building energy patterns," *Energies*, vol. 6, pp. 579-597, 2013.
- [16] C. Cassisi, P. Montalto, M.A. Aliotta, A. Pulvirenti, "Similarity measures and dimensionality reduction techniques for time series data mining," in: A. Karahoca (Ed.), *Advances in Data Mining Knowledge Discovery and Applications*, Chapter 3, IntechOpen, London, 2012, pp. 71-96.
- [17] D. Berndt, J. Clifford, "Using dynamic time warping to find patterns in time series," *Workshop on KDD*, vol. 10, no. 16, Seattle, USA, July 31 - August 01, 1994, pp. 359-370.
- [18] M. Müller, *Information Retrieval for Music and Motion*, Springer-Verlag, Berlin, 2007, 318 p.
- [19] N. Begum, L. Ulanova, J. Wang, E. Keogh, "Accelerating dynamic time warping clustering with a novel admissible pruning strategy", *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, Australia, August 10-13, 2015, pp. 49-58.
- [20] S. Chu, E.I. Keogh, D. Hart, M. Pazzani, "Iterative deepening dynamic time warping for time series", *Proceedings of the 2-nd SIAM International Conference on Data Mining*, Arlington, USA, April 11-13, 2002. pp. 195-212.
- [21] A. Zinke, D. Mayer, *Iterative Multi Scale Dynamic Time Warping*, Universität Bonn, Technical Report CG-2006-1, 2006, 11 p.
- [22] T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, Berlin, 1995, 364 p.
- [23] T. Kohonen, "Essentials of the self-organizing map," *Neural Networks*, vol. 37, Elsevier, pp. 52-65, 2013.



**Mashtalir Sergii**, graduated from Kharkiv National University of Radio Electronics (KhNURE) in 2001 (M.Sc.), got his PhD degree in 2005 and Dr. Habil. Sci. Eng. in 2016. Currently working as a Professor at Informatics department in KhNURE. The author of about 70 scientific publications including 2 monographs. The interests of scientific research include but not limited to image processing and recognition, video parsing, content based image and video retrieval.



**Mikhnova Olena**, graduated from Kharkiv National University of Radio Electronics in 2008 with Masters Degree in Information Systems and Technologies. In 2012 graduated from Kharkiv University of Humanities People's Ukrainian Academy, obtained Translation and Interpreting specialty there. In 2014 gained PhD degree in Artificial Intelligence. Currently working as a lecturer assistant at Kharkiv Petro Vasilenko National Technical University of Agriculture, Cybernetics department. Translator and co-author of nearly 10 published books in the field of Computer Science for the most part, which is the area of scientific R&D.



**Stolbovyi Mykhailo**, graduated from Kharkiv National University of Radio Electronics in 2014, obtained Masters degree in System Engineering. PhD student at Informatics department of the same university by now. Among the scientific interests are image and video processing, video clustering, and computational intelligence.