Time Series Mining in a Psychological Domain

Vladimir Kurbalija
Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg D. Obradovica 4, 21000 Novi Sad, Serbia
+381 21 4852877
kurba@dmi.uns.ac.rs

Hans-Dieter Burkhard
Institute of Informatics
Humboldt University Berlin, Germany
Rudower Chaussee 25, 12489 Berlin
0049-30-20933160
hdb@informatik.hu-berlin.de

Mirjana Ivanović
Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg D. Obradovica 4, 21000 Novi Sad, Serbia
+381 21 4852855
mira@dmi.uns.ac.rs

Charlotte Meyer
Institute of Psychology
Humboldt-Universität zu Berlin, Germany
Rudower Chaussee 18, 12489 Berlin
0049-30-20939328
charlotte.meyer@hu-berlin.de

Jens Nachtwei
Institute of Psychology
Humboldt University Berlin, Germany
Rudower Chaussee 18, 12489 Berlin
0049-30-20939373
jens.nachtwei@hu-berlin.de

Lidija Fodor
Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg D. Obradovica 4, 21000 Novi Sad, Serbia
lidyxy@yahoo.com

ABSTRACT

Analysis of time series became an inevitable tool in many application areas, such as stock market analysis, process and quality control, observation of natural phenomena, medical treatments, etc. The domain of psychology also offers some interesting applications for time-series analysis, while it is not so frequently applied recently. In this paper, we utilize the system for time-series analysis FAP (developed at Department of Mathematics and Informatics, University of Novi Sad) on the data gained from a specific experimental lab system (so called Socially Augmented Microworld – SAM) developed by informatics and psychologists for Human Factors Research at Humboldt University Berlin. On the basis of experiment log files we extracted three types of time series and generated distance matrices using three kinds of time series similarity measures. Finally, we performed clustering on generated distance matrices and produced dendrograms which serve as the basis for deeper analysis. The outcome of this analysis is two-folded: (a) the most suitable similarity measure can be selected for this domain and (b) these results can serve as a basis for the development of artificial agents which may replace human participants in the experiment.

Categories and Subject Descriptors

J.4 **[SOCIAL AND BEHAVIORAL SCIENCES]**, *Psychology.*

I.5.3 Clustering, *Similarity measures*.

I.2.4 Knowledge Representation Formalisms and Methods

General Terms

Algorithms, Experimentation, Human Factors.

Keywords

Time-series mining, FAP, Microworld, social interaction.

1. INTRODUCTION

Time series represent the way how measured data change over time, often defined as a sequence of data points, typically measured at regular intervals [1]. The use and manipulation of time series in scientific research has noticeably grown in recent years, as the interest for understanding the principles underlying time series data increased [5]. Analysis of time series is a useful tool to detect dynamic patterns in large datasets within a particular time range in order to make forecasts [2]. However time series analysis encompasses a variety of methods, a common one being time series data mining (TSDT), a unification of traditional data mining and forecasting techniques based on time dimensions and predictive analyses [3]. Data mining refers to a nontrivial process of identification and separation of valid, authentic and potentially useful and understandable patterns and relationships. Major fields of interest related to mining time series data are indexing, classification, clustering, predicting or forecasting, summarization, anomaly detection and segmentation [1, 4, 5, 6, 7, 8]. For most of these fields there are several important concepts which have to be considered when processing and analysing time series: pre-processing transformation, time series representation and similarity or distance measure. According to this, the team from the Department of Mathematics and Informatics, University of Novi Sad, has developed a multipurpose, multifunctional tool set: FAP – Framework for Analysis and Prediction [9]. FAP is a framework which supports all three important concepts with the possibility to easily change some existing or to add new concrete implementation of any concept.

Research on time series presented in this paper is based on an interdisciplinary project of the Institute of Informatics and Psychology at Humboldt University Berlin, Germany [10]. At the heart of this project is a specific experimental lab system, containing a so called Socially Augmented Microworld (SAM), which is described in more detail in section 3. Logfile data obtained from prior experiments with this lab system are used for time series analysis via FAP. As these data are based on regular measurements they represent an excellent ground for conducting different techniques of time series analyses to investigate which kind is most suitable to represent actual patterns of behaviour and actual behavioural differences in reality, respectively. Results will serve as a basis for the development of artificial agents, which will be "fed" with data of human behavioural patterns to react similarly to humans and thus may replace human participants in the experiment.

To examine the data we read the values from the logfiles, and created time series on that basis first. We used the FAP system for representing, manipulating and computing similarities between these time series. On the basis of very rich experiment logfiles, we decided to create three types of time series: path series, acceleration series and deviation series. All these kinds of series will be explained in more detail in section 4. Based on mentioned time series, we generated different distance matrices, using three kinds of similarity measures (see section 2). Finally, we applied clustering on created distance matrices, to further examine the data and try to find some reasonable explanations for performed experiments.

The rest of this paper is organized as follows: Section 2 gives an explanation about time series related to the FAP system. The experiment, as a base for time series data, is described in Section 3. Section 4 explains further work on creating time series, generating distance matrices based on them, and performing clustering on resulting matrices. Finally, Section 5 concludes the paper and gives the possibilities for further work.

2. TIME SERIES AND FAP

Time-series analysis and mining has become a very popular research area in the past decade and as a consequence a huge amount of proposed techniques and algorithms have been developed. However, several main concepts must be distinguished when dealing with time-series data: *similarity measures*, *representation* and *pre-processing transformation*.

Similarity measures provide an efficient way to determine for two time series data whether and in what extent they are similar. Finding a semantically appropriate definition of time series similarity measures is challenging. The central task is to compare different time series to find out whether different time series show in some sense similar behaviour. These comparisons are usually based on shapes and patterns. Many useful distance measures for this purpose are proposed in the literature: Euclidean distance (ED) [4], Dynamic Time Warping (DTW) [12], Longest Common Subsequence distance (LCS) [13], Edit Distance with Real Penalty (ERP) [14], Edit Distance on Real Sequence (EDR) [15], Sequence Weighted Alignment model (Swale) [16] etc.

Although a distance function can be defined and computed, the similarity model should still allow for imprecise matches. That is the main reason for the importance of pre-processing transformations, which can help to resolve different distortions in the raw data. Also, it is very important to find the suitable way to represent time series data on a high level, in order to downsize the dimensionality of time series while keeping its original shape.

FAP (Framework for Analysis and Prediction) is a framework for time series analysis, designed to support the main concepts for working with time series data. As a multipurpose, multifunctional, extendable and freely available system, it supports significant techniques for mining time series data. FAP includes the possibility of using different representations of time series, main similarity measures and required preprocessing tasks. One of the greatest advantages of the framework is a consolidation of main aspects related to time series mining in one place, and the ease way of combining them. As mentioned, the system is open-source, and implemented in Java programming language, to ensure easy maintenance and future upgrades. The framework can provide a great help to researchers in testing and comparing newly introduced concepts with already existing ones.

FAP ensures that all the leading concepts related to mining time series are covered, and is organized through ten packages. The main package of the system is *fap.core*. This package contains several packages which provide implementations for data points, lists of data points, series, some additional mathematical concepts, representations, similarity measures and other useful supporting features. Common similarity measures are implemented into the system, including: Lp, DTW, CDTW (Constraint DTW), LCS, CLCS (Constraint LCS), ERP, EDR, Swale. All the provided properties contributed to the convenience of representing and mining time series data on one place.

FAP is already successfully applied on testing the influence of global constraints of DTW and LCS similarity measures [17], while the investigation of global constraints for EDR and ERD measures is in progress.

3. THE SAM EXPERIMENT

3.1 Short description of the SAM experiment

The data used for time series analysis with FAP were obtained from experiments with a specific lab system developed by informatics and psychologists for Human Factors Research at Humboldt University Berlin. Altogether the system consists of two components, (1) a microworld and (2) an operator supervising this microworld. Previous studies were conducted mainly to study performance of operators in complex process control settings (Nachtwei & Meyer, submitted [18]). However in this paper performance of SAM only is relevant to the actual research objective so it is described in more detail.

In Human Factors Research microworlds are commonly used to model complex and dynamic reality. Compared to traditional microworlds the so called *Socially Augmented Microworld* (SAM) developed at Humboldt University Berlin, consists not only of a technical but of a social component with two participants (navigators) representing actual parts of the microworld. These two participants perform a cooperative tracking task. Both are given a joystick to steer an object along a virtual track on a monitor placed in front of them. Each joystick contributes to 50% of vertical and horizontal motion of the object, hence they are forced to do the steering cooperatively. In addition, both navigators are instructed in a different way (however as they are not allowed to communicate they assume having been given the same instruction). Navigator 1 is told to focus on high speed while navigating whereas the Navigator 2 is instructed to emphasize on accuracy. These different instructions induce a goal conflict which aims to enhance the complexity of SAM (making it more comparable to real complex and dynamic systems). The cooperative tracking of the two navigators is in turn supervised by a third participant, an operator who is asked to support the two other participants, i.e. improve their steering performance, by using information provided by a specific interface. Steering performance of the subsystem SAM as well as supervisory control performance of the operator is logged every 39 ms. The logfiles analysed in this paper are based on performance data of SAM only (two participants doing their cooperative tracking task alone without being supported by a third person). The idea is to observe the two human navigators steering the object first individually (solo mode) and then in cooperation with the other navigator over a number of trials (cooperative mode) (see next section).

Analysing steering behaviour of these navigators allows to draw conclusions for different fields, e.g. in terms of behaviour based personality, patterns of social interaction or for creating artificial agents based on these patterns, which may replace human navigators in future experiments. Regarding the development of artificial agents, a lot of additional factors have to be considered in order to achieve agents to act similarly to humans. However the very first step is the analysis of human behaviour and the possibility to identify patterns of behaviour in specific situations. An earlier paper [10] reports on an analysis done by human visual inspection for the solo mode. A number of typical navigator types could be identified based on specific behavioural patterns, e.g. adapted navigator, extreme steering navigator, no steering navigator, parallel navigator, jittering navigator. On this basis, a very simple agent performing reactive proportional control was developed which showed similar behaviour to the human "jittering navigator" type [11]. However, the manual analysis of typical behaviours appeared to be very time consuming. Hence, the objective is to find better methods for analysis and clustering using the automated comparison of time series. In the future related patterns will be used to make agents react in a way typical for humans.

3.2 The structure of log files

In total, 26 test sessions (data sets) were included in time series analysis, each data set containing single as well as cooperative performance of navigators. A team of two navigators was assigned to each session which in turn consisted of 11 trials in total. The first four of these 11 trials are solo trials (with each navigator steering alone) named "learning track (navigator 1)", "test track (navigator 1)", "learning track (navigator 2)", "test track (navigator 2)". These four solo trials are followed by three cooperative trials of navigators as well as four additional trials where a supervising operator was included. In this paper we considered only the first four steps of each experiment (Results obtained from all other steps of experiments will be investigated in future research activities). Thus, logfiles of single participant performance on trials 1 to 4 (i.e. each participant navigating the object alone with a joystick of 50% input) were extracted from an overall file containing data sets of prior experiments. Each trial describes a track of certain length and complexity. In the beginning of a test session tracks are short and simple, i.e. no curves or obstacles have to be managed. In the end of a test session tracks are quite long and consist of curves, obstacles and forks, so a lot of decision making is necessary for both navigators in these trials. The intervals of measurements of steering behaviour are 39 seconds for all steps of all test sessions. One trial is represented by a CSV file, containing all the necessary data about periodical measurements of the trial. Most relevant variables for Time Series Data Mining were: *TimeTotal, TimeDelta, XCoordinateOnSection, YCoordinateOnSection, DistanceTotal, DistanceTick*. The variable *TimeTotal* represents total elapsed time from the beginning of the observed step of the experiment in milliseconds. *TimeDelta* is a time value showing the duration of one observation, the interval between the previous and the current measuring in milliseconds. *XCoordinateOnSection* and *YCoordinateOnSection* are current values of x and y coordinates on the path. *DistanceTotal* is a value of total travelled distance from the beginning of the step, and the *DistanceTick* column shows the crossed distance between the previous and current measuring. An example of a logfile is shown in Figure 1.

TimeTotal	TimeDelta	XCoordinate	YCoordinate	ColorCode	DistanceTotal	DistanceTick
40	40	387	110.24	88	10.24	10.24
79	39	387	120.48	88	20.48	10.24
118	39	387	130.72	88	30.72	10.24
157	39	387	140.96	88	40.96	10.24
196	39	387	151.2	88	51.2	10.24
235	39	387	161.44	88	61.44	10.24
275	40	387	171.68	88	71.68	10.24
314	39	386.35	182.49	88	82.49	10.81
353	39	385.69	193.62	88	93.62	11.13
392	39	384.83	204.91	88	104.91	11.29

Figure 1. A part of CSV file, representing the first step of the first experiment

4. MINING SAM TIME SERIES DATA

4.1 The creation of time series

As was highlighted before, based on experimental data three types of time series were created for each trial of a test session: path series, acceleration series and deviation series. Data points of all three series contain two coordinates. One coordinate of the data point which is constant over all three series is the time value, which represents the total elapsed time from the beginning of the experimental observation sequence, or in other words from the beginning of the observed step. Time value is represented by the *TimeTotal* column from the CSV file. The second coordinate of the data point varies over the three series.

For the path series, the second coordinate is a value of the distance covered since the last measuring, which is the value of the *distanceTick* field from the CSV file. In the case of acceleration series, the second coordinate indicates whether there has been an acceleration or deceleration in the current measuring, compared to the previous measure. We calculated this value as a simple difference between the distance taken in the current and in the previous measuring. Finally, the second coordinate of data points in deviation series shows the deviation from the ideal trajectory on the path. The ideal motion on the track is the shortest possible way to cross the path. For both of the trials ("learning track" and "test track"), there already existed a list of CSV files representing the ideal motion on certain parts of the paths. In each of these parts coordinates were relative to that part, instead of to the whole path. We matched these part files into one instance of *Serie*, to represent the ideal trajectory for the path. Afterwards, for values of *y* coordinates of the step, we found the appropriate *x* coordinate values in series for ideal motion, and compared them with *x* coordinates in the step file (simply counted as difference between *x* coordinate values). This way the deviation of the motion on the path from the ideal one was determined.

4.2 Generating distance matrices

The created time series represented the starting point for further experiments and the base for running similarity measures and creating distance matrices. Among available similarity measures, implemented in the FAP system, we selected DTW, EDR and ERP similarity measures to create appropriate distance matrices, mainly because these measures support matching time series of different lengths. In this domain, all time series are of different length because various navigators complete the same track in different times.

The main idea and motivation for further experiments with obtained time-series data was to compare all the generated series of one type for both tracks, separately. Three types of series (path, acceleration and deviation series) had been created for the first four trials of every test session (two trials of "learning track", another two of "test track"). Thus, all three types of series were compared regarding both "learning track" as well as "test track" over all 26 sessions. Since two navigators are assigned to every test session, 52 path, (as well as 52 acceleration and 52 deviation series respectively) resulted each for "learning track" as well as for "test track", so the distance matrices have a dimension of 52. Consequently 6 appropriate distance matrices were created.

With regard to the application of three different similarity measures for these types of series, 3 x 6 = 18 distance matrices resulted in total (DTW, EDR, ERP matrices for "learning track" path series; DTW, EDR, ERP matrices for "learning track" acceleration series; DTW, EDR, ERP matrices for "learning track" deviation series; DTW, EDR, ERP matrices for "test track" path series; DTW, EDR, ERP matrices for "test track" acceleration series; DTW, EDR, ERP matrices for "test track" deviation series).

An example of a DTW distance matrix for path series for "learning track", is shown in Table 1. On the diagonal, all the values are 0, because at that point, similarity measure was used to compare a series with itself. As DTW calculates distances between time series, the relation between some time series can be easily seen in table 1.

Table 1. A part of CSV file with DTW distance matrix, for path series for "learning track"

Experim1navigator1	Experim1navigator2	Experim2navigator1
0	18008.35	2759.605
18008.35	0	7192.057
2759.605	7192.057	0
5916.756	3472.793	3530.143
6758.235	4829.235	3382.454

4.3 Mining Time Series, Clustering, Dendrograms, Analysis

The existing CSV files containing distance matrices represent a fruitful starting point for applying clustering algorithm. Clustering or cluster analysis describes the procedure of assigning a set of objects into groups (called clusters), so that the objects in the same cluster are more similar to each other than to those in other clusters [1]. Clustering helps to group data, and determine some patterns and behaviour of the data. Clustering is a suitable data mining technique in cases when the data set is already defined, and general patterns need to be determined and recognized from the data, like it is the case here.

Various clustering algorithms can significantly vary in their properties, and different cluster models can be used in algorithms. Some typical cluster models are: connectivity models, centroid models, distribution models, density models, subspace models, group models. Also, we can roughly distinguish two big groups of clustering: hard clustering and soft clustering. Hard clustering means that each object belongs to a cluster or not, while soft or so called fuzzy clustering means that each object belongs to every cluster to a certain degree.

Within these experiments we used the method of hierarchical clustering for created distance matrices. Hierarchical clustering seeks to build a hierarchy of clusters. Strategies for hierarchical clustering can be generally divided into two types: agglomerative and divisive [1].

The results of hierarchical clustering are usually presented in form of dendrogram, and it is important that different linkage criteria can be used. The linkage criterion determines the distance between sets of observations as a function of the pairwise distances between observations. Some commonly used linkage criteria between two sets of observations are maximum or complete linkage clustering, minimum or single linkage clustering, mean or average linkage clustering (or UPGMA) and minimum energy clustering. On that base, we generated dendrograms, using Mathlab, to graphically represent hierarchical clustering. An example for the result of this kind of clustering is given in Figure 2.

This figure shows the clustering results of ERP similarity measures for path series based on the trial „learning track". By looking at this dendrogram several conclusions can be drawn. For example, it can be seen which navigators behave similarly on this track considering taken path and according to ERP similarity measure. Furthermore, all the navigators can be grouped in any desired number of groups by "cutting-off" the dendrogram on the desired level. Regardless of the number of groups, this approach guaranties the groups with the most similar navigators inside one group.

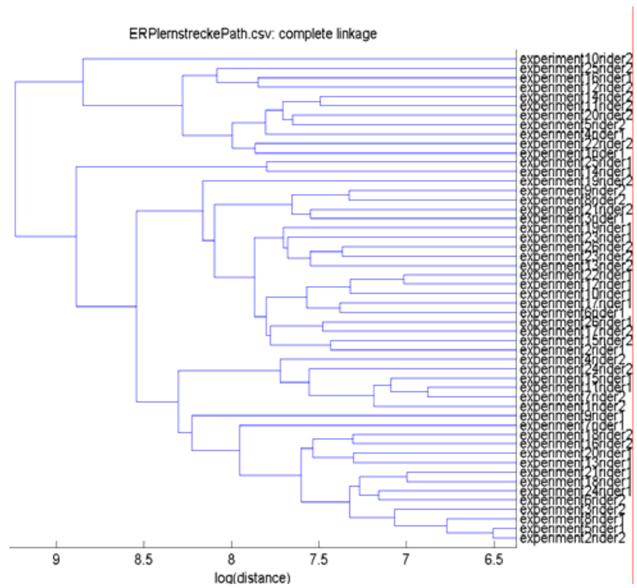

Figure 2. A dendrogram for hierarchical clustering, for ERP distance matrix for path series on "learning track, complete linkage

Three similarity measures (DTW, EDR, ERP) were implemented in the FAP framework. Each measure follows a completely different methodology and ends up with different results of clustering. Therefore, we are especially interested in which of the three measures is the most suitable for this application, i.e. creates distinct clusters which in the future can be meaningfully interpreted as specific navigator types (e.g. those identified in [11]). However, in a first approach the three measures were compared based on a more simple classification: As navigators are instructed in a different way (navigator 1 focussing on speed vs. navigator 2 focussing on accuracy), this should result in different navigating styles regarding path, deviation and acceleration series. Thus similarity measures should return in clusters representing each the group of speeding navigators vs. the group of accurate navigators. Initial descriptive analyses of all 18 dendograms indicate that ERP measure seems most appropriate to group cases of speed vs. accurately steering navigators over all three series path, acceleration and deviation. Resulting clusters contained either a much larger proportion of navigators 1 vs. navigators 2 respectively thus separating speedy from accurate navigators. Thus, the analysis of generated dendrograms represents a promising starting point for further inferential analysis.

5. CONCLUSION AND FURTHER WORK

FAP – Framework for Analysis and Prediction supports three important concepts, namely pre-processing transformation, time series representation and similarity or distance measure. FAP was applied to further analyse logfile data from a cooperative tracking study based on a specific experimental lab system (SAM). Logfiles of navigating behaviour of two differently instructed participants (focus on speed vs. accuracy) served as a basis for creating time series, which were then explored by cluster analysis. To this end, DTW, EDR and ERP were selected as three different similarity measures and implemented in the FAP system. Of particular interest is the question, which of these three similarity measures may be most appropriate to identify two distinct clusters, which distinguish navigators focussing on speed from those focussing on accuracy. Descriptive analysis of 18 dendograms revealed ERP measure to be most suitable to represent cluster speedy vs. accurate navigators. However differences are significant only in some cases and further analysis of time series via multivariate analysis is required. What is more, as three kinds of measures were tested over three kinds of series, a decision criterion (which measure is appropriate for which series?) is difficult to draw. Furthermore, a complete two cluster solution (grouping all of the 52 time series in two clusters) has not been identified. This could be due to the fact, that the mere instruction of navigators to drive with a focus on speed vs. accuracy leaves them too much freedom to follow this instruction. Thus the instruction may have affected only some, but not all navigators, consequently not every time series can be clearly assigned to one group or the other.

With regard to the framework itself, this approach proved that FAP can serve as a promising tool for the experimental domain. It was applied successfully and with low effort as only the preparation of time-series data, but no modifications of the framework itself are necessary.

A challenging future task is the comparison of human navigators with agents of different complexity. In the solo mode human navigators follow a very simple control rule (reactive proportional control) as reported in [10] and [11]. In cooperative mode navigators are confronted with conflicting tasks (speed vs. accuracy) and thus adaptation and learning behaviour will be shown by human navigators. Thus, for agents performing similarly, a more complicated architecture will be needed. It requires a memory to store and to compare control with results (which depend on the partner's activities). Moreover certain learning strategies will have to be realized by the architecture. However it is questionable, which learning strategies are suitable to model human behaviour. Thus the experiments and the time series analysis may lead to new insights concerning models of human behaviour.

In this paper, not only the potential of FAP and its competences themselves could be validated, but data obtained in experiments with SAM could be analysed on a much richer base via FAP as complete series of logfile data could be compared. Thus FAP provides a much deeper understanding of individual (and cooperative) behaviour and can contribute to an improved modelling of software agents and their behaviour. Consequently FAP will be used in the future systematically to analyse actual behavioural patterns in even more complex settings, i.e. in settings when the operator comes into play. The objective will be to analyse time series of interactive behaviour of navigators themselves as well as navigators being supported by an operator.

6. ACKNOWLEDGMENTS

The authors gratefully acknowledge DAAD for support through the bilateral project "Intelligent Techniques for Data Integration and Decision Support in the Medical Domain" (2011–2012). V. Kurbalija and M. Ivanović, thank the Ministry of Education and Science for support through project no. OI174023, "Intelligent Techniques and their Integration into Wide-Spectrum Decision Support".

7. REFERENCES

[1] J. Han and M. Kamber. *Data Mining: Concepts and Techniques.* Morgan Kaufmann Publishers, CA, 2005.

[2] Y. Zhang, B. J. Jansen and A. Spink. Time series analysis of a Web search engine transaction log. *Information Processing and Management 45*, 2009, pp. 230–245.

[3] Schubert & Lee (2011)

[4] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. Fast Subsequence Matching in Time-Series Databases. *In SIGMOD Conference*, 1994, pp. 419-429.

[5] E. J. Keogh. A Decade of Progress in Indexing and Mining Large Time Series Databases. *In VLDB*, 2006, pp. 1268-1268.

[6] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, E. Keogh. Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measures. *In VLDB '08*, 2008, Auckland, New Zealand, pp. 1542-1552.

[7] C. A. Ratanamahatana, J. Lin, D. Gunopulos, E. Keogh, M. Vlachos and G. Das, *Data mining and knowledge discovery handbook 2010*, Part 6, 1049-1077, DOI=10.1007/978-0-387-09823-4_56

[8] D. Pfoser,Y. Tao, K. Mouratidis, M. A. Nascimento, M. Mokbel, S. Shekhar and Y. A. N. HUANG, *Advances in Spatial and Temporal Databases*, 12th International Symposium, SSTD 2011, Minneapolis, MN, USA, August 24-26, 2011. Proceedings.

[9] V. Kurbalija, M. Radovanović, Z. Geler and M. Ivanović. A framework for time-series analysis. *In Proceedings of the 14th international conference on Artificial intelligence: methodology, systems, and applications* (AIMSA'10), 2010, pp. 42-51.

[10] H-D. Burkhard, L. Jahn, S. Kain, C. Meyer, J.Muetterlein, J. Nachtwei, N. Niestroj, S. Rougk and M. Schneider. Artificial Subjects in the Psychological Experiment "Socially Augmented Microworld (SAM)". *Proceedings of International Workshop CS&P 2011*, 28 -30 September 2011, Pułtusk, Poland, pp. 54-66.

[11] L. Jahn and M. C. Schneider. Dokumentation der Entwicklung von Agenten zur Ersetzung von Mikroweltbewohnern in der Socially Augmented Microworld. Studienarbeit an der Humboldt University Berlin, 2012.

[12] E. J. Keogh and C. A. Ratanamahatana. Exact indexing of dynamic time warping. *Knowl. Inf. Syst.*, 7(3), 2005, pp. 358-386.

[13] M. Vlachos, D. Gunopulos, and G. Kollios. Discovering similar multidimensional trajectories. *In ICDE*, 2002, pp. 673-684.

[14] L. Chen and R. T. Ng. On the marriage of lp-norms and edit distance. *In VLDB*, 2004, pp. 792-803.

[15] L. Chen, M. T. ¨Ozsu, and V. Oria. Robust and fast similarity search for moving object trajectories. *In SIGMOD Conference*, 2005, pp. 491-502.

[16] M. D. Morse and J. M. Patel. An efficient and accurate method for evaluating time series similarity. *In SIGMOD Conference*, 2007, pp. 569-580.

[17] V. Kurbalija, M. Radovanović, Z. Geler and M. Ivanović. The influence of global constraints on DTW and LCS similarity measures for time-series databases. *In Proceedings of S3T'11, 3rd International Conference on Software, Services and Semantic Technologies*, volume 101 of Advances in Intelligent and Soft Computing, Bourgas, Bulgaria, 2011, pp. 67–74.

[18] Nachtwei, J. & Meyer, C. (submitted). Between keyhole and clutter effect - A multi-level evaluation of interface extensions using the ATEO master display (AMD).