Symbolization of time-series: An evaluation of SAX, Persist, and ACA

Anita Sant'Anna
School of Information Science
Computer and Electrical Engineering,
Halmstad University - Sweden
Email: anita.santanna@hh.se

Nicholas Wickström
School of Information Science
Computer and Electrical Engineering,
Halmstad University - Sweden
Email: nicholas.wickstrom@hh.se

***Abstract*—Symbolization of time-series has successfully been used to extract temporal patterns from experimental data. Segmentation is an unavoidable step of the symbolization process, and it may be characterized on two domains: the amplitude and the temporal domain. These two groups of methods present advantages and disadvantages each. Can their performance be estimated *a priori* based on signal characteristics? This paper evaluates the performance of SAX, Persist and ACA on 47 different time-series, based on signal periodicity. Results show that SAX tends to perform best on random signals whereas ACA may outperform the other methods on highly periodic signals. However, results do not support that a most adequate method may be determined *a priory*.**

I. Introduction

Symbolic time-series analysis (STA) has been successfully used in many different application areas to identify temporal patterns in experimental data [1]. Although simple dynamics may be observed with traditional analytical tools such as Fourier Transforms, symbolization can improve the analysis of processes that are complex and possibly chaotic. Symbolization may also reduce sensitivity to noise and greatly improve computational efficiency [1]. Symbolization of time-series also allows for the use of techniques developed for symbolic data, e.g. data mining in databases [2], data mining on DNA sequences [3], text mining [4], knowledge representation and reasoning [5], and computational linguistics [6].

For the purpose of STA the following symbolization properties are of interest:

- Alphabet size: Ideally, the symbolic representation of a signal contains the minimum number of symbols needed to express all its underlying dynamics. In reality, since these dynamics are usually unknown, choosing an alphabet size is a mostly empirical task and should consider the trade off between information loss and the complexity of the analysis.
- Information loss: Symbolization will always incur some loss of information. The best results are achieved when the information lost is superfluous to the analysis at hand, e.g. noise.
- Compression: The improvement in computational complexity achieved by symbolization is due to the compression of the symbolized signal, where an interval of data is represented by one single symbol.
- Temporal information: The underlying hypothesis of STA is that symbolization simplifies the signal but retains its temporal characteristics, enabling the discovery of its dynamics.

An unavoidable part of the symbolization process is segmentation. Time-series segmentation can be described in two domains: the amplitude or the temporal domain. We will refer to the former as quantization, and to the latter as temporal segmentation. Figure 1 shows a very simple example of each type of segmentation and subsequent symbolization.

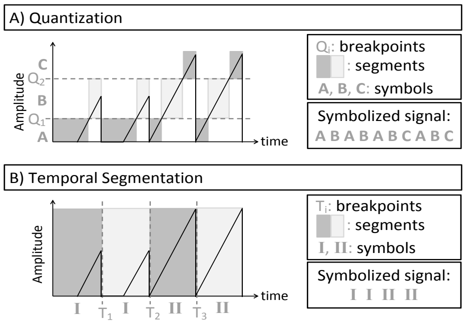

Fig. 1. **Quantization vs. Temporal Segmentation.** This simple example illustrates the main differences between quantization (A) and temporal segmentation (B). The breakpoints of quantization are in the amplitude domain, and each interval may be assigned a symbol. The breakpoints of temporal segmentation are in the temporal domain, and similar segments must be clustered before they can be assigned a symbol.

The breakpoints of quantization techniques can be obtained in a variety of ways, e.g. equidistant intervals, mean ± standard deviation, equiprobable intervals (SAX) [7], enduring states (Persist) [8]. Symbolization after quantization is very straight forward, an unique symbol is assigned to the segments in each quantization interval.

Temporal segmentation, on the other hand, can be achieved using fixed-size windows, zero crossings of the signal and its derivative [9], signal variance [10], clustering (ACA) [11], piecewise linear approximation [12], among others. Symbolization after temporal segmentation depends on grouping similar segments into classes. A symbol can then be assigned to each class.

Numerous segmentation techniques are available, so how can the best symbolization method be chosen? Is quantization

978-1-4244-9306-7/11/$26.00 ©2011 IEEE

better than temporal segmentation? Are there intrinsic data characteristics which work best with one class of method or another? Given the nature of quantization and temporal segmentation, one possible hypothesis is that signals presenting clear recurring patterns can be well symbolized by temporal segmentation methods, whereas signals with no temporal structure are better symbolized by quantization methods.

This paper investigates whether the periodicity of the signal may be used to determine *a priori* which type of method is advised. Two quantization methods and one temporal segmentation method were chosen, namely Symbolic Aggregate Approximation (SAX), Persist, and Aligned Cluster Analysis (ACA). Their performance was evaluated on 47 different signals, ranging from random to periodic, synthetic and real, of various sizes. Each method was evaluated in terms of information loss and compression factor. An estimate of signal periodicity was used to characterize signals and investigate if the performance of each method is dependent on certain signal properties.

II. Related Work

As mentioned previously, there are many possible ways to segment and symbolize a signal. This paper investigates SAX, Persist and ACA. SAX was chosen for it has been shown to perform very well in many application areas. The breakpoints of SAX, however, are independent of signal characteristics. Persist, on the other hand, designs breakpoints that are optimized to certain signal characteristics. In addition, unlike breakpoints based on signal statistics, Persist can create a varied number of segments. The chosen representative for temporal segmentation was ACA. This was the most promising method described in the literature that combined both segmentation and clustering. The remaining of this section elaborates on each of these methods.

A. SAX

Symbolic Aggregate Approximation (SAX) [7] is a symbolic representation of time-series based on the Piecewise Aggregate Approximation (PAA) representation [13], and the assumption that time-series are normally distributed. SAX can reduce a time-series of length N to a symbolic string of length W ($W < N$) composed of Z different symbols ($Z > 2$). Figure 2 exemplifies SAX.

This method is characterized by two important advantages:

- Dimensionality reduction: The dimensionality reduction achieved with PAA [13] is also present in SAX.
- Lower bounding of distance measure: It has been shown that a distance measure between two symbolic strings created by SAX lower bounds the true distance between the two original time-series [7].

An experimental validation of SAX [7] investigated its symbolic representation applied to several data mining problems: hierarchical clustering, partitional clustering, nearest neighbor classification, decision tree classification, query by content, detection of anomalous behavior, motif discovery, and visualization aspects. SAX's performance was compared to SDA [14] and IMPACTS [15]. The analysis concluded that SAX was "competitive with, or superior to, other representations on a wide variety of data mining problems" [7].

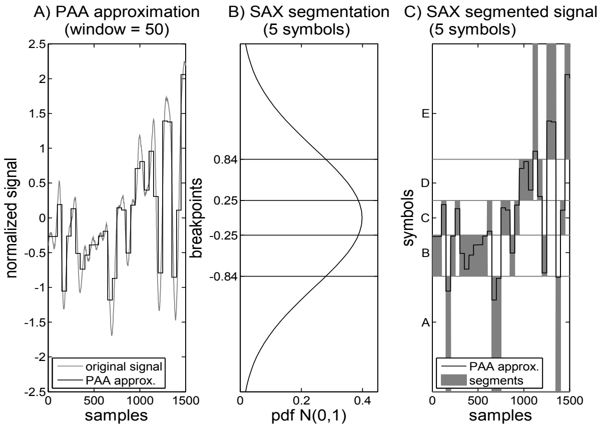

Fig. 2. **SAX symbolization.** (A) shows the normalized original signal and its PAA approximation for a window size of 50 samples. (B) shows the breakpoints for alphabet size $Z = 5$. (C) illustrates how the PAA values in between breakpoints are assigned symbols.

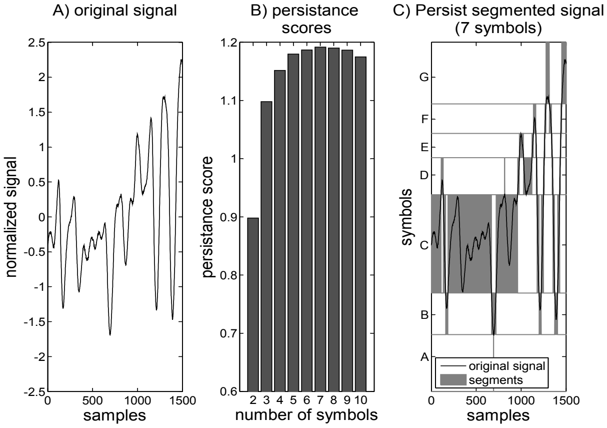

Fig. 3. **PERSIST symbolization.** (A) shows the normalized original signal. (B) illustrates the persistence scores obtained for each number of symbols. The best score is obtained with 7 symbols. (C) shows the optimal segmentation with 7 symbols.

B. Persist

Persist is an unsupervised discretization method [8] designed to identify relevant enduring states in time-series. This method segments the signal so as to maximize the persistence of each symbol. The main objective of Persist is to devise breakpoints that somehow relate to the underlying temporal characteristics of the signal. The method uses self-transition probabilities of each symbol as an indicator of its persistence. Figure 3 illustrates how Persist works.

The performance of Persist was compared to eight other methods on both artificial and real data [8]. The methods investigated were: equi-probable symbols based on signal distribution, SAX, equi-distant symbols over the data range, mean ± standard deviation, median ± adjusted median absolute deviation, k-means, Gaussian Mixture Model, and Hidden Markov Model. These methods and Persist were applied to artificial and real data with known enduring states. Persist was found to outperform the other methods on artificial data, and to create more meaningful breakpoints on real data [8].

C. ACA

Aligned Cluster Analysis (ACA) [11] is a method for unsupervised clustering of temporal patterns in human motion data. The main difficulties in segmenting human motion data stem from large intra-person variability, wide range of temporal scales, irregularity in the periodicity of human action and the exponential nature of possible movement combinations [11].

The method copes with these issues by:

- Enabling user control over the temporal scale of the actions of interest.
- Providing a robust distance metric based on Dynamic Time Alignment Kernel (DTAK) [16].
- Formulating the segmentation problem as an energy minimization problem, which can be solved with an efficient coordinate descent algorithm.

ACA is an extension of kernel k-means clustering [17] that allows for variable numbers of features in each observation and uses a distance metric (DTAK) robust to noise and invariant to the speed of the action. Because of computational complexity, it is impractical to rum ACA on large amounts of data, and temporal reduction might be required. In addition, the performance of the algorithm depends on adequate initialization. Dedicated temporal reduction and initialization methods for motion capture data were introduced [11].

This method was tested on both synthetic 1-dimensional data and real motion capture data. The method was shown promising but further work is needed to "automatically select the optimal number of actions and avoid local minima in the optimization" [11]. Figure 4 exemplifies the use of ACA.

III. Method

A. Information Loss

Information loss was estimated by the Mean Absolute Error (MAE) between original signal and reconstructed signal after symbolization.

For SAX and Persist, a symbol value was estimated for each symbol by taking the average of all original samples corresponding to that symbol. The reconstructed signal was created by substituting each original sample with its corresponding symbol value.

For ACA, after segmentation/clustering a template was created for each symbol. The segments belonging to the same cluster were time shifted so as to be best aligned with each other, and the ensemble average was taken. The reconstructed signal was formed by substituting each segment for its template. Because the segments may be of different lengths, the template was re-sampled to adjust to the length of the segment.

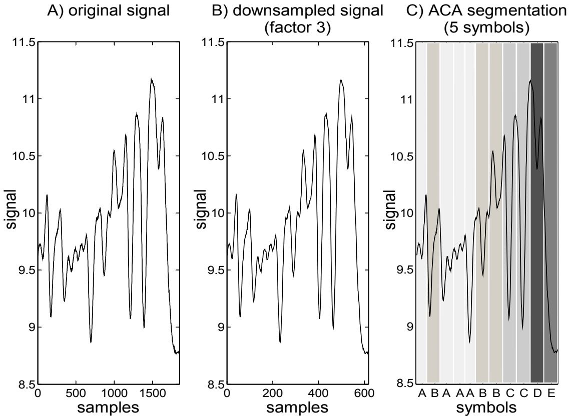

Fig. 4. **ACA symbolization.** (A) shows the original signal. Because no dedicated temporal reduction methods have been developed for arbitrary data, the signal was simply down-sampled by a factor of 3. The down-sampled signal is shown in (B). (C) illustrates the resulting segmentation and how segments are grouped into symbols.

B. Compression Factor

Each symbolized signal was composed of: 1) a symbolic string, ***Symb***, of size N; 2) a vector of size N containing the duration of each symbol, ***Dur***; and 3) a matrix of size ZxP containing symbol values or templates, ***Templ***. The compression factor was calculated as:

$CompFac = |Bits_{Orig} - Bits_{Comp}|/Bits_{Orig}$

where:

$Bits_{Comp} = Bits_{Symb} + Bits_{Dur} + Bits_{Templ}$
$Bits_{Orig} = sizeof(double) * M = 64 * M$
$Bits_{Symb} = ceil(log2(Z)) * N$
$Bits_{Dur} = ceil(log2(max(\boldsymbol{Dur})) * N$
$Bits_{Templ} = sizeof(double) * Z * P$

and M is the length of the original signal, $ceil$ rounds upwards, Z is the number of symbols, N is the length of the symbolic string, and $P = 1$ for SAX and Persist.

C. Periodicity

The periodicity measure indicates if there are significant periodic patterns in the data. Since no gradual scale of periodicity was found in the literature, it was estimated by the maximum peak value (excluding the zero lag peak) of the normalized auto-correlation of the detrended signal.

D. Data

Forty-seven 1-dimensional signals were used. These signals were small extracts of the following public data sets:

- [1]Synthetically generated control charts [18]: The first instance of the Normal class.
- [1]UCR Time Series Classification database [19]: One arbitrarily chosen instance from each of the available 19 data sets.

[1] Data provided by Eamon Keogh.

- walk8-gait data set [20], [21]: The first 4000 samples of the X-axis acceleration values.
- [1]ECG signals from PhysioBank Archives [22]: One example from each of the following MIT-BIH databases: Normal Sinus Rhythm, Malignant Ventricular Arrhythmia, Supraventricular Arrhythmia.
- [1]Personal income estimates from the (U.S.) Bureau of Economic Analysis [23]: Personal income estimates for the state of California from 1929 to 1999.
- [1]Inline skating EMG signals [8], [24]: The signal relating to the activation of the Medial Gastrocnemius muscle.
- The CMU Multi-Modal Activity Database [25]: Extracts from Subject 07 cooking brownies and eggs, the first axis of Sparkfun IMU data from arms and legs.
- CMU Graphics Lab Motion Capture Database [26]: Extracts from Subject2 performing the activities: walk, run and jog, jump and balance, punch and strike, bend over, swordplay, and wash self. The first dimension of the data matrix was used.
- [1]Population estimates from the U.S. Census Bureau [27]: The population of the U.S from 1900 to 1999.
- [1]Stock market data [28]: The daily opening prices of an arbitrary company traded in the New York Stock Exchange from 26-Mar-90 to 24-Oct-03.

In addition to these data, the following were generated:

- The previously explained Walk8 data, filtered with a 5-sample window mean filter.
- The CMU Motion Capture dataset punch and strike data repeated (concatenated) 4 times.
- A fractional Brownian motion signal with Hurst parameter 0.7, and length of 600 samples.
- A 600-sample long sinusoidal signal of amplitude one and period of 63 samples.

E. Analysis

The SAX algorithm was downloaded from the SAX homepage [29]. The symbolization function requires: the original signal, ***Orig***, of length M; the length of the PAA window, w; the size of the desired symbolic representation n; and the number of symbols, Z. For simplicity, $w = n = M$ were used, resulting in a symbolic string of length M. The consecutive occurrences of the same symbol were then combined into one symbol entry i and stored in ***Symb***(i), and the corresponding number of consecutive occurrences of the symbol were stored in ***Dur***(i). The signal was then reconstructed, the information loss and the compression factor calculated as explained in Parts III-A and III-B respectively. This analysis was undertaken considering, for each file, number of symbols varying from $Z = 2$ to $Z = 15$.

Similarly, the Persist algorithm was downloaded from the homepage [30]. To calculate the persistence scores and the optimized breakpoints, the function requires: the original signal; and the minimum, $minNumSymb$, and maximum, $maxNumSymb$, number of symbols to consider. In order to obtain results for several different numbers of symbols, $minNumSymb = maxNumSymb = Z$ was used. Each sample of the original signal was then assigned a symbol, according to breakpoint intervals. Consecutive occurrences of the same symbol were combined into one symbol entry i, stored in ***Symb***(i), and the corresponding number of consecutive occurrences of the symbol were stored in ***Dur***(i). The signal was then reconstructed and the information loss and compression factor calculated for each file, considering $Z = 2$ through $Z = 15$.

The ACA algorithm was downloaded from the homepage [31]. Given the impracticalities of analyzing long signals, the signals longer than 600 samples were down-sampled by $factor = round(M/600)$. The function requires the following inputs: the original signal; the minimum, $minSizeSeg$, and maximum, $maxSizeSeg$, segment lengths; and the number of symbols, Z. After segmentation, for each segment i, its symbol, ***Symb***(i), and its length, ***Dur***(i), are stored. The signal was reconstructed as explained in Part III-A, and the information loss and compression factor were calculated. For each file, $Z = 2$ through $Z = 10$ were considered.

The performance of ACA is very sensitive to $minSizeSeg$ and $maxSizeSeg$. In order to find the best values for each file, the analysis explained above was undertaken for all combinations of $minSizeSeg$ varying from 5 to 100, and $maxSizeSeg$ varying from $minSizeSeg + 10$ to 300. The values resulting in the smallest information loss were chosen. The analysis was then repeated 10 times with the chosen values, and the best results were used in the analysis.

The analysis was undertaken in MATLAB (MathWorks, Natick, MA).

IV. Results

The periodicity value (Part III-C) was calculated for each signal. The signals were grouped according to periodicity as shown in Table I. Most of the signals presented low periodicity values, [0, 0.2]. As expected, the Brownian motion signal falls within this group. Signals known to be more periodic, such as the walk8 data set, presented higher values, (0.4, 0.6]. The sinusoid signal belongs to the highest interval, (0.6, 0.9]. Notice that the CMUmocap punch and strike signal, jumps from the first interval to the last when repeated 4 times (CMUmocap repeated). No signals presented periodicity value higher than 0.9.

Figure 5 exemplifies some of the characteristics of reconstructed signals for each of the methods. Notice that extreme values of the original signal, i.e. peaks, are not captured by SAX because they lie at the edge of the distribution; nor by Persist because they are not enduring values. On the other hand, ACA may perform extremely well (Figure 5(A)), and adequately characterize peaks and valleys. On occasion, the reconstructed patterns may be misaligned with the original signal (Figure 5(B)), resulting in high information loss values. This happens because the reconstruction of the signal is very sensitive to ACA segmentation, which in turn, is very sensitive to initial conditions.

The results of information loss versus compression factor are displayed by periodicity interval in Figure 6. A directly

Periodicity interval:[0, 0.2]		Periodicity interval:(0.2, 0.4]	
signal	**length (samples)**	**signal**	**length (samples)**
CMUmocap jump and balance	483	ECG venarh	1000
CMUmocap punch and strike	1854	income	71
CMUmocap run and jog	173	CMUmocap walk	343
CMUmocap swordplay	1500	stocks	3421
CMUmocap wash self	2645	UCR coffee	286
Population	100	UCR face all	131
UCR 50words	270	UCR face four	350
UCR beef	470	UCR gun point	150
UCR CBF	128	UCR OSU leaf	427
UCR lighting2	637	UCR olive oil	570
UCR lighting7	319	UCR yoga	426
UCR trace	275	**Periodicity interval: (0.4, 0.6]**	
UCR two patterns	128	CMUmocap bend over	2235
UCR synthetic control	60	control chart	60
UCR wafer	152	UCR Adiac	176
CMUkitchen right leg brownie	27180	UCR fish	463
CMUkitchen right leg eggs	25657	UCR Swedish leaf	128
CMUkitchen right arm brownie	27191	walk8	4000
CMUkitchen right arm eggs	25676	walk8 filtered	4000
CMUkitchen left leg brownie	27221	**Periodicity interval: (0.6, 0.9]**	
CMUkitchen left leg eggs	25699	ECG normal	512
CMUkitchen left arm brownie	27216	ECG superven	512
CMUkitchen left arm eggs	25705	inline skating	29900
Brownian motion	600	CMUmocap repeated	7416
		Sine	600

TABLE I
PERIODICITY. SIGNALS GROUPED ACCORDING TO PERIODICITY.

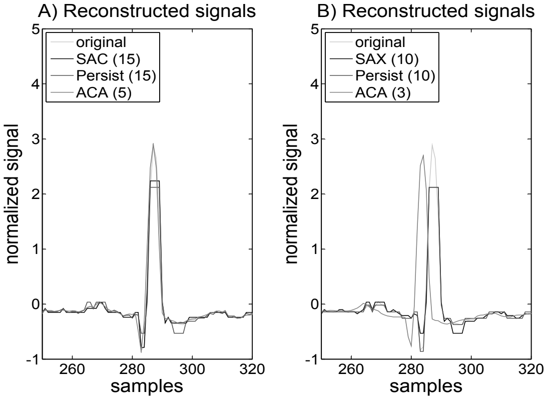

Fig. 5. **Reconstructed signals**. Examples of reconstructed signals for each of the methods, the number of symbols used is expressed in parenthesis. (A) shows an example where ACA was very successful. (B) shown an example where the ACA segmentation caused the reconstructed templates to be misaligned with the original signal. The original signal is a normal ECG data. The figures shows only a detail of the signal.

proportional trend is observed, i.e. higher compression factors result in higher information loss. For SAX and Persist it is clear that more symbols result in lower information loss and lower compression. ACA, on the other hand, is less predictable. Notice also that the lowest compression achieved increases with periodicity. For the first periodicity interval, compression varies from around 0.65 to 1. In the highest periodicity interval, compression only varies between 0.9 and 1. This indicates that signals with periodic patterns are symbolized more efficiently. From Figure 6 it is observed that ACA can, at times, perform very well, and at times, considerably poorer than the other two methods, especially for low periodicity signals. This apparent inconsistency may be due to the reconstruction method's sensitivity to segmentation, and the segmentation's sensitivity to initial conditions. It may also reflect ACA's inadequacy for segmenting random signals.

It is clear from Figure 7 that, for low periodicity signals, SAX may outperform the other methods. A small portion of ACA results outperformed Persist but mostly, ACA is confined to high-compression, high-loss sectors. These signals present no striking underlying structure, which makes it difficult for Persist to find enduring states, and for ACA to find good segment clusters. As the periodicity of the signal increases, the best ACA results improve, the best SAX results worsen, and the best Persist results stay unchanged around 0.05 information loss. In the highest periodicity interval, ACA outperforms the other methods for one particular signal, the sinusoid.

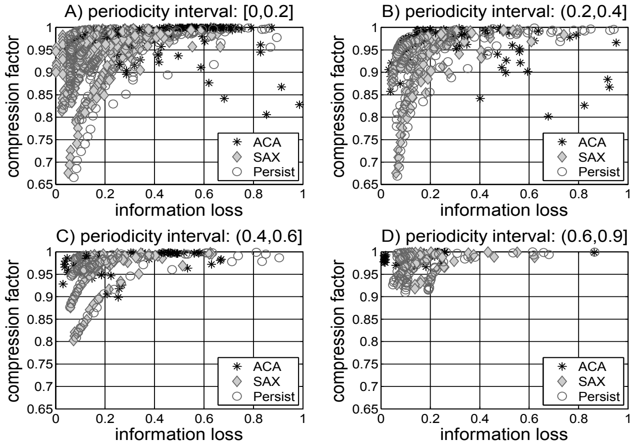

Fig. 6. **Information loss versus compression factor**. Results for files contained in each periodicity interval. For each file, SAX and Persist results encompass $Z = 2, 3, ..., 15$, and ACA results include $Z = 1, 2, ..., 10$. For each of these, the resulting information loss is plotted against the achieved compression factor.

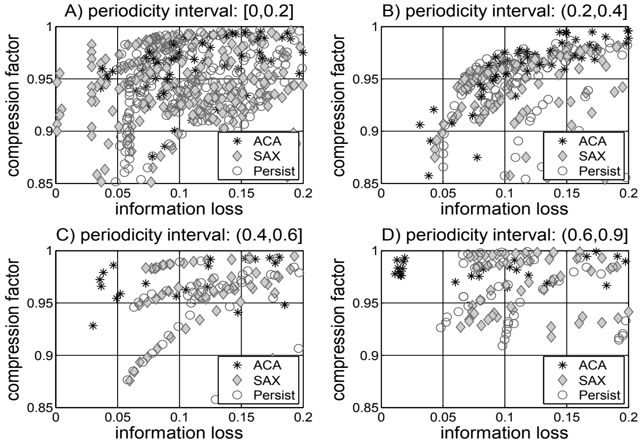

Fig. 7. **Detail of Figure 6**. The plots presented previously were zoomed in on compression values 0.85 to 1, and information loss values 0 to 0.2.

Although the number of signals in the first interval is much superior to the number of signals in the other intervals, certain tendencies can be observed for each method depending on the periodicity characteristics of the signal. However, the majority of results overlap and there is no clear indication of which method would perform best given the signal's periodicity, therefore decisions must be made on a case by case basis.

One final observation is necessary, Persist and ACA were not designed for symbolization of arbitrary time-series. Persist is based on the assumption that the data contains enduring underlying states, and ACA was designed for temporal segmentation of mocap data. Nonetheless, their application to arbitrary time-series was shown possible, and in alignment with the characteristics expected from quantization and temporal segmentation methods.

V. Conclusion

This paper investigated the symbolization of time-series based on quantization and temporal segmentation. The objective of the analysis was to determine if the performance of each method could be predicted based on signal characteristics. The hypothesis was that quantization methods are more appropriate for signals with no temporal structure, whereas temporal segmentation is advised when the signal presents periodic patterns.

Two quantization methods, namely SAX and Persist, and one temporal segmentation method, ACA, were considered. Methods were evaluated based on information loss and compression factor. Forty-seven varied signals were considered, they were divided into four groups according to their periodicity. Results indicated that there is a tendency for SAX to perform better on non-periodic signals, and for ACA to perform better on periodic signals. However, the majority of results overlapped and an *a priori* evaluation of signal periodicity cannot determine which method is more appropriate.

Although these results were not statistically conclusive, they support the idea that certain signal characteristics influence the performance of different symbolization methods. Further work is needed to develop measures of signal properties which may predict the performance of time-series symbolization methods.

Acknowledgment

The authors would like to thank Eamon Keogh for providing most of the data used in the experiment. The authors of SAX, Persist and ACA are much appreciated for providing MATLAB implementations of their methods, especially Feng Zhou for his help with the ACA temporal reduction problem. Some of the data used in this paper was obtained from kitchen.cs.cmu.edu and the data collection was funded in part by the National Science Foundation under Grant No. EEEC-0540865.

References

[1] C. S. Daw, C. E. A. Finney, and E. R. Tracy, "A review of symbolic analysis of experimental data," *Review of Scientific Instruments*, vol. 74, no. 2, p. 915, 2003.

[2] H.-H. Bock and E. Diday, *Analysis of symbolic data : exploratory methods for extracting statistical information from complex data*. Springer, 2000.

[3] K.-S. Leung, K. H. Lee, J.-F. Wang, E. Ng, H. Chan, S. Tsui, T. Mok, P.-H. Tse, and J.-Y. Sung, "Data mining on dna sequences of hepatitis b virus," *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, vol. 8, no. 2, pp. 428 –440, 2011.

[4] W. Fan, M. Gordon, and P. Pathak, "Discovery of context-specific ranking functions for effective information retrieval using genetic programming," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 4, pp. 523 – 527, 2004.

[5] N. Leone, G. Pfeifer, W. Faber, T. Eiter, G. Gottlob, S. Perri, and F. Scarcello, "The dlv system for knowledge representation and reasoning," *ACM Transactions on Computational Logic*, vol. 7, pp. 499–562, 2006.

[6] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvisticae Investigationes*, vol. 30, pp. 3–26.

[7] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing SAX: a novel symbolic representation of time series," *Data Mining and Knowledge Discovery*, vol. 15, pp. 107–144, 2007.

[8] F. Mörchen, A. Ultsch, and O. Hoos, "Extracting interpretable muscle activation patterns with time series knowledge mining," *International Journal of Knowledge-based and Intelligent Engineering Systems*, vol. 9, pp. 197–208, 2005. [Online]. Available: http://portal.acm.org/citation.cfm?id=1233864.1233868

[9] G. Guerra-Filho and Y. Aloimonos, "A language for human action," *Computer*, vol. 40, no. 5, pp. 42–51, 2007.

[10] A. Zinnen, K. V. Laerhoven, and B. Schiele, "Toward recognition of short and non-repetitive activities from wearable sensors," in *AmI-07: European Conference on Ambient Intelligence*, 2007, pp. 142–158.

[11] F. Zhou, F. Torre, and J. Hodgins, "Aligned cluster analysis for temporal segmentation of human motion," in *8th IEEE International Conference on Automatic Face Gesture Recognition*, 2008, pp. 1–7.

[12] A. Sant'Anna and N. Wickström, "A symbol-based approach to gait analysis from acceleration signals: Identification and detection of gait events and a new measure of gait symmetry," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 5, pp. 1180 – 1187, 2010.

[13] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra, "Locally adaptive dimensionality reduction for indexing large time series databases," *SIGMOD Record*, vol. 30, pp. 151–162, 2001.

[14] H. André-Jönsson and D. Badal, "Using signature files for querying time-series data," in *Principles of Data Mining and Knowledge Discovery*, ser. Lecture Notes in Computer Science, J. Komorowski and J. Zytkow, Eds. Springer Berlin / Heidelberg, 1997, vol. 1263, pp. 211–220.

[15] Y.-W. Huang and P. S. Yu, "Adaptive query processing for time-series data," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1999, pp. 282–286.

[16] H. Shimodaira, K.-I. Noma, M. Nakai, and S. Sagayama, "Dynamic time-alignment kernel in support vector machine," in *Advances in Neural Information Processing Systems*, vol. 2, 2001, pp. 921–928.

[17] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[18] D. Pham and A. Chan, "Control chart pattern recognition using a new type of self organizing neural network," *Journal of Process Mechanical Engineering*, pp. 115–127, 1998.

[19] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana, "The UCR Time Series Classification/Clustering Homepage," http://www.cs.ucr.edu/~eamonn/time_series_data, 2006.

[20] K. Van Laerhoven and A. Aronsen, "Memorizing what you did last week: Towards detailed actigraphy with a wearable sensor." in *27th International Conference on Distributed Computing Systems Workshops*, IEEE Computer Society. IEEE Computer Society, 2007, p. 47.

[21] "Darmstad technical university, embedded sensing systems - walk8-gait dataset," http://www.ess.tu-darmstadt.de/datasets/walk8-gait.

[22] "physiologic signal archives for biomedical research," http://www.physionet.org/physiobank/database/.

[23] "Bureau of economic analysis - regional economic accounts," http://www.bea.gov/regional/spi/.

[24] F. Mörchen and O. Hoos, "Inline skating EMG data," Philipps-University Marburg, Germany.

[25] "The Quality of Life Grand Challenge Data Collection: Kitchen," http://kitchen.cs.cmu.edu/main.php.

[26] "Cmu graphics lab motion capture database," http://mocap.cs.cmu.edu/.

[27] "U.s. census bureau," http://www.census.gov/population/.

[28] Y. Cai and R. Ng, "Indexing spatio-temporal trajectories with chebyshev polynomials," in *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*. ACM, 2004, pp. 599–610.

[29] "Symbolic aggregate approximation homepage," http://www.cs.ucr.edu/~eamonn/SAX.htm.

[30] "Persist time series discretization," http://www.mybytes.de/persist.php.

[31] "Aligned cluster analysis's software," http://www.humansensing.cs.cmu.edu/aca/code.html.