



**HAL**  
open science

## Bag of recurrence patterns representation for time-series classification

Nima Hatami, Yann Gavet, Johan Debayle

► **To cite this version:**

Nima Hatami, Yann Gavet, Johan Debayle. Bag of recurrence patterns representation for time-series classification. *Pattern Analysis and Applications*, 2019, 22 (3), pp.877 à 887. 10.1007/s10044-018-0703-6 . hal-01774237

**HAL Id: hal-01774237**

**<https://hal.science/hal-01774237>**

Submitted on 22 Oct 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Bag of Recurrence Patterns Representation for Time-Series Classification

Nima Hatami · Yann Gavet · Johan Debayle

Received: date / Accepted: date

**Abstract** Time-Series Classification (TSC) has attracted a lot of attention in pattern recognition, because wide range of applications from different domains such as finance and health informatics deal with time-series signals. Bag of Features (BoF) model has achieved a great success in TSC task by summarizing signals according to the frequencies of "feature words" of a data-learned dictionary. This paper proposes embedding the Recurrence Plots (RP), a visualization technique for analysis of dynamic systems, in the BoF model for TSC. While the traditional BoF approach extracts features from 1D signal segments, this paper uses the RP to transform time-series into 2D texture images and then applies the BoF on them. Image representation of time-series enables us to explore different visual descriptors that are not available for 1D signals and to treat TSC task as a texture recognition problem. Experimental results on the UCI time-series classification archive demonstrates a significant accuracy boost by the proposed Bag of Recurrence patterns (BoR), compared not only to the existing BoF models, but also to the state-of-the art algorithms.

**Keywords** Time-Series Classification (TSC) · Bag of Features (BoF) representation · visual words · dictionary · Recurrence Plots (RP).

## 1 Introduction

A time-series is a sequence of data points (measurements) which has a natural temporal ordering. Many important real-world pattern recognition tasks deal with time-series analysis. Biomedical signals (e.g. EEG and ECG), financial data (e.g. stock market and currency exchange rates), industrial de-

---

N. Hatami, Y. Gavet, J. Debayle

The SPIN center, École Nationale Supérieure des Mines de Saint-Étienne, France

Tel.: +33 472437543

E-mail: hatami@creatis.insa-lyon.fr

N. Hatami is currently with the CREATIS center, INSA-Lyon and University of Lyon, France

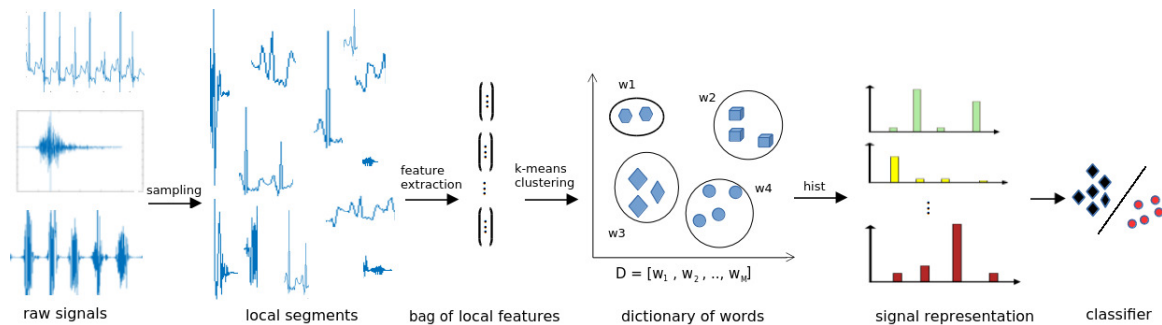


Fig. 1: The traditional BoF model for time-series classification.

vices (e.g. gas sensors and laser excitation), biometrics (e.g. voice, signature and gesture), video processing, music mining, forecasting and weather are examples of application domains with time-series nature [49, 57, 8, 24, 52]. The time-series mining motivations and tasks are mainly divided into curve fitting, function approximation, prediction and forecasting, segmentation, classification and clustering categories. In a univariate time-series classification,  $x^n \rightarrow y^n$  so that  $n$ -th series of length  $l$ :  $x^n = (x_1^n, x_2^n, \dots, x_l^n)$  is associated with a class label  $y^n \in \{1, 2, \dots, N_c\}$ . It is worth noting that although this paper is mainly focused on time-series classification problem, the proposed method can be easily adapted to the other tasks such as clustering and anomaly detection.

The existing time-series classification methods may be categorized from different perspectives. Regarding the feature types, "frequency-domain" methods include spectral analysis and wavelet analysis; while "time-domain" methods include auto-correlation, auto-regression and cross-correlation analysis. Regarding the classification strategy, it can also be divided into "instance-based" and "feature-based" methods. The former measures similarity between any incoming test sample and the training set; and assigns a label to the most similar class (the euclidean distance based 1- Nearest Neighbor and Dynamic Time Wrapping (DTW) are two popular and widely used methods of this category [26, 27, 38, 48, 53]). The latter first transforms the time-series into the new space and extract more discriminative and representative features in order to be used by a pattern classifier which aiming to find optimum classification boundaries [18, 15, 36, 40, 39].

The BoF (also known as bag-of-words) is a simplifying representation that originally used in natural language processing and information retrieval. In this model, a text (such as a sentence or a document) is represented as the bag (multiset) of its words, disregarding grammar and even word order but keeping multiplicity. The BoF is commonly used for document classification where the (frequency of) occurrence of each word is used as a feature for training a classifier. Later, the BoF model has also adopted for computer vision and signal processing. Recently, BoF model has achieved a high recognition rate for time-series classification [7, 49, 30, 4, 51]. The main idea is to represent a

signal by a histogram of feature words of a data-learned dictionary (codebook). There are five steps to perform BoF model: i) creating local segments from the input signals, ii) extracting local features from the segments and creating the "bag", iii) learning codebook, iv) quantizing features using codebook and representing samples by histogram of words, and v) classification. Fig.1 shows the traditional BoF model for time-series classification.

Traditionally, when applying BoF on a series, features are extracted from "bag of segments" obtained by sliding a window over the signal. On the way towards exploring different discriminative and informative features, the proposed work aims to investigate an alternative approach devoted for time-series analysis, by transforming 1D signals into 2D images before applying BoF. This provides extra tools for addressing time-series analysis by taking advantage of existing advanced image processing and computer vision methods. Beside the currently available signal processing measures and statistics that traditionally being used for time-series feature extraction, the proposed approach provides access to different efficient features and descriptors such as Scale-invariant feature transform (SIFT)[31], Histogram of Oriented Gradients (HOG) [12] and Local Binary Patterns (LBP)[37] which are defined only in image space.

There are various approaches for visualization of time-series with the advantage of better understanding and analyzing them [28]. This paper investigates using a Recurrence Plot (RP) [16] - an advanced technique of nonlinear data analysis in chaos theory to visualize time-series - for TSC task. The RP is the visual inspection of higher dimensional phase space trajectories and exhibits global characteristic (*typology*) and local patterns (*texture*). Instead of quantization of RP matrix with measures such as recurrence rate, determinism, mean diagonal line length and entropy as traditionally suggested, Vinicius et al. [46] propose to treat plots as gray-level texture images and apply an SVM classifier on texture features. In this work instead, we propose a Bag of Recurrence Patterns (BoR) model by applying a popular method of BoF on vocabulary of local image patterns. The experimental results on the publicly available UCI time-series dataset [10] demonstrate that treating the time-series classification problem as image texture recognition boosts the accuracy by extracting more representative and discriminative information using features and descriptors which are only applicable in image space.

## 2 Related Work

This section briefly reviews the recent contributions on time-series classification using BoF model, novel texture image recognition techniques and application of the RP on TSC task.

Recently, there has been a big effort to improve and adapt BoF for time-series application. Multiple subsequences selected from random locations and of random lengths suggested by Baydogan et. al. [7] to handle the time warping of patterns. Mel-Frequency Cepstrum Coefficients (MFCC) features are used in [17] for music genre classification and artist identification. Vectors of

Discrete Wavelet Transform (DWT) features from each of the local segments is used in BoF framework for EEG and ECG classification [49]. Grabocka et al. [19] build words using coefficients of local polynomial approximations and use their histogram to classify repetitive time series. Bailly et al. [4] adapted SIFT image descriptors for 1D domain. A BoF approach using statistical features is applied to identify general motion primitives for modeling different human activities [57]. Bromuri et al. [8] combine BoF and kernel methods-based dimensionality reduction to perform multi-label time-series classification on health records of chronically ill patients. In another research, a method for discovering characteristic patterns in a time series based on Symbolic Aggregate approxImation and Vector Space Model (SAX-VSM) is proposed by [44]. Raw signals are converted into bags of words using SAX; followed by calculation of term frequency-inverse document frequency (TF-IDF). Class labels are assigned using maximal cosine similarity. Wang and Oates [52] inspired by the max-pooling approaches in Convolutional Neural Networks (CNN) and extend SAX to multivariate synchronous vital signs data. Finally, [51] proposed a time warping SAX in order to integrate the temporal correlation. Instead of sampled randomly or uniformly, [58] used a peak and valley detection algorithm and subsequences are sampled from these peaks and valleys. Then, Histogram of Oriented Gradients (HOG-1D) and Dynamic time warping-Multidimensional scaling (DTW-MDS), are proposed to describe segments. Bag-of-SFA-Symbols (BOSS) ensemble [43] is a noise-resistant method that combines Symbolic Fourier Approximation (SFA) representation and BoF model. Learned Pattern Similarity (LPS)[6] is an unsupervised approach to represent and measure the similarity and models the dependency structure in time series that generalizes the concept of auto-regression to local autopatterns.

Image texture recognition is one of the important tasks in computer vision and image processing. Here, a short review of some texture recognition methods are given. Multiscale rotation-invariant sampling based texture image classification is proposed in [13]. A multiscale wavelet transform is used to decompose the magnitude pattern (MP) mapping of a texture and the sampled directional mean vectors (SDMVs) of each wavelet subband is computed. In addition, frequency vectors (FVs) of the sign pattern mappings are also constructed for capturing the structural information of textures. The texture information at a given scale is described by three SDMVs and an FV. A texture classification approach is proposed by modeling joint distributions of local patterns with Gaussian mixture models (GMM) [29]. A given texture region is filtered by a set of filters and subsequently the joint probability density function is estimated by GMMs. A statistical approach to represent the marginal distribution of the wavelet coefficients using finite mixtures of generalized Gaussian (MoGG) distributions is proposed [2]. The MoGG captures a wide range of histogram shapes, which provides better description and discrimination of texture. Dense Micro-Block Difference (DMD) [35] local features are proposed for texture images. The local features are then projected into a low dimensional space using Random Projections and the low dimensional features are encoded into a image descriptor using Fisher vectors. In another research,

energy features are extracted from shearlet (extension of wavelets) subbands and their dependences are modelled using linear regression [14]. The regression residuals are used for defining the distance from a texture image to a class.

Although RP [16] is proposed a long time ago, application of RP on TSC has not been fully explored. In fact, there are not so many types of approaches, when it comes to application of time-series images on TSC. Traditionally, PRs are described by Recurrence Quantification Analysis (RQA) measures, which quantifies the number and duration of recurrences of a dynamical system presented by its state space trajectory [56,33]. Roma et al. [42] explored the use of the RQA features for auditory scene classification. These features are computed over a thresholded similarity matrix computed from windows of MFCC features. Added to traditional MFCC statistics, they improve accuracy. Their experiments suggest that RQA features have some discriminating power with respect to auditory scenes, that is not captured in basic MFCC statistics. In another application, RQA parameters are extracted from the RP in order to classify the EEG signals into normal, epileptic (ictal) and background (pre-ictal) classes [1]. Yang [55] and Chen and Yang [11] applied RQA on Vectorcardiogram (VCG) and heart rate variability (HRV) signals in multiple wavelet scales and investigated different classifiers for the identification of cardiac disorders (discrimination between Myocardial Infarction (MI)/congestive heart failure (CHF) and healthy control (HC) subjects). The multiscale RQA are found to be adequately discriminatory to capture the dynamic variations between HC/CHF and MI. Silva et al. [45] measured the similarity between RPs using Campana-Keogh (CK-1) distance, a Kolmogorov complexity-based distance that uses video compression algorithms to estimate image similarity. They show that RP allied to CK-1 distance lead to accuracy improvements compared to Euclidean distance and DTW on the UCR data sets. In another work by Silva et al. [46] image texture features i.e. LBP, Grey Level Co-occurrence Matrix (GLCM), Gabor filters, and Segmentation-based Fractal Texture Analysis (SFTA) are extracted from RPs and then, TSC task is addressed by applying a SVM classifier on combination of those features. Taking advantage of the Convolutional Neural Networks (CNN)'s high performance on image classification, Hatami et al. [20] first transformed time-series signals into texture images (using RP) and then classify them by a 2-layer CNN model. Obtaining high recognition rates on the UCR datasets demonstrates the efficiency of the proposed pipeline using RP features and CNN classifiers on TSC.

### 3 Methodology

Time-series can be characterized by a distinct recurrent behavior such as periodicities and irregular cyclicities. Additionally, the recurrence of states is a typical phenomena for dynamic non-linear systems or stochastic processes that time-series are generated in [41,34,47]. The RP is a visualization tool that aiming to explore the m-dimensional phase space trajectory through a

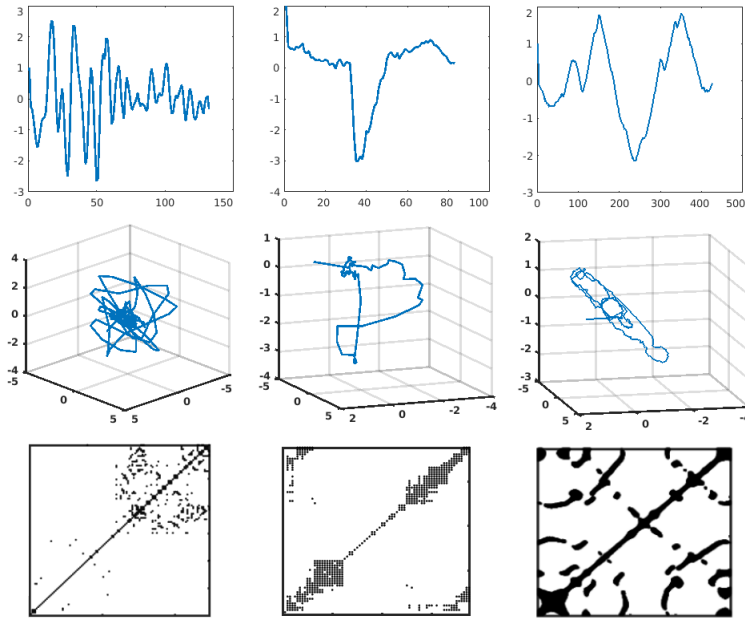


Fig. 2: Application of RP ( $m = 3$  and  $\tau = 4$ ) on three different datasets from the UCR time-series classification archive [10]: left) FaceAll, middle) TwoLeadECG, and right) Yoga data (from left to right, respectively). First row is the signal plot, second row demonstrates the 3D space trajectory and last row visualize the R-matrix (traditional binary plots using  $\theta = 0.5$ ).

two-dimensional representation of its recurrences. The main idea is to reveal in which points some trajectories return to a previous state and it can be formulated as:

$$R_{i,j} = \theta(\epsilon - \|\mathbf{s}_i - \mathbf{s}_j\|), \quad \mathbf{s}(\cdot) \in \mathfrak{X}^m, \quad i, j = 1, \dots, K \quad (1)$$

where  $R_{i,j}$  are elements of the RP matrix,  $m$  is dimension of the phase space trajectory,  $N$  is the number of considered states  $\mathbf{s}$ ,  $\epsilon$  is a threshold distance,  $\|\cdot\|$  a norm and  $\theta(\cdot)$  the Heaviside function (a.k.a the unit step function whose value is zero for negative argument and one for positive argument). There is also another important parameter involve in generating the RP matrix: embedding time delay  $\tau$  that can be considered as sub-sampling factor of the time-series signal. The R-matrix contains both texture which are single dots, diagonal lines as well as vertical and horizontal lines; and typology information which characterized as homogeneous, periodic, drift and disrupted. For instance, fading to the upper left and lower right corners means the process contains a trend or drift; or vertical and horizontal lines/clusters show that some states do not change or change slowly for some time and this can be interpreted as laminar states [32]. Application of RP (using the phase space

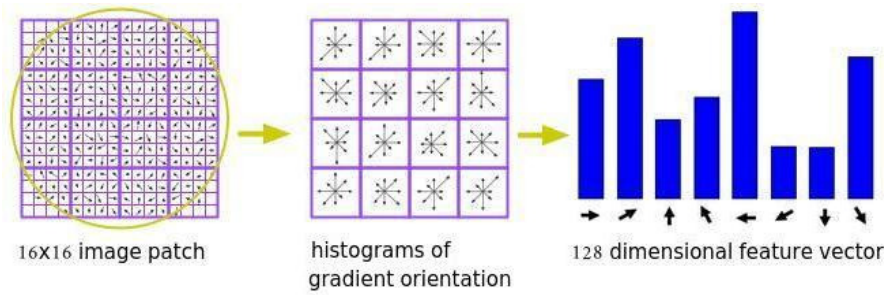


Fig. 3: The SIFT descriptor generation. First the gradient magnitude and orientation at each image sample point in a  $16 \times 16$  patch is computed (left). These samples are then accumulated into orientation histograms summarizing the contents over  $4 \times 4$  sub regions. The length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region (middle). The final 128 ( $16 \times 8$ ) dimensional feature vector is obtained by concatenating the histograms (right).

dimension  $m = 3$  and embedding time delay  $\tau = 4$  parameters) on three different exemplar datasets from the UCR time-series classification archive [10] e.g. FaceAll, TwoLeadECG and Yoga data is shown in Figure 2.

Inspired by the unique texture images obtained from the R-matrices, this paper proposes a time-series classification pipeline based on the BoF model. First the raw 1D time-series signals  $x^n$  are transformed into 2D recurrence texture images, and then the local patches are extracted using blob detectors (here, we applied multi-scale dense grid sampling). The SIFT is a robust and popular computer vision algorithm to detect and describe local features in images (more details on the SIFT descriptor are given in 3). In the proposed BoR model, SIFT operator is applied to describe  $K$  patches of an image  $S = [s_1, s_2, \dots, s_K] \in \mathfrak{R}^{128 \times K}$  (since there are  $4 \times 4 = 16$  histograms each with 8 bins the vector has 128 elements). Then, descriptors are used as the input for the k-means clustering algorithm in order to generate the *initial* dictionary of length  $M$ ,  $D_{init} = [w_1, \dots, w_M]$ , where  $w_i$  is the  $i$ th initial visual word and optimum  $M$  is determined experimentally and depends on the problem. The k-means clustering aims to partition samples into  $M$  clusters in which each sample belongs to the cluster with the nearest mean, serving as a prototype of the cluster.

There is a big effort in the BoF and dictionary learning literature for improving the quality of the codebooks. The proposed BoR model uses an online incremental codebook optimization method, proposed by [50], based on the Locality-constrained Linear Coding (LLC). Instead of the traditional BoF image representation using histogram of words, the main idea of the LLC is to project each descriptor into its local-coordinate system, followed by max-pooling combined with  $\ell^2$  normalization as suggested in [54], to generate the feature vectors. LLC applies locality constraint to select similar basis of local



image descriptors from a codebook, and learns a linear combination weight of these basis to reconstruct each descriptor. For ease of reference, the codebook optimization process is shown in Algorithm 1. After optimizing the initial dictionary,  $D_{opt} = [w_1^*, \dots, w_M^*]$ , the coding process is done in 3 steps: i) finding k-nearest words of SIFT descriptor  $s_i$  denoted as  $D_i$ , ii) reconstructing  $s_i$  using  $D_i$  by  $\tilde{c} = \text{argmax}_c \|s_i - D_i c\|^2$ , s.t.  $\sum_j^k c_j = 1$ , which  $c_i$  is a code for  $S$ . One of the advantages of the LLC coding is that classification of its representation (features) using linear classifier obtains better results compared to the traditional spatial pyramid matching (SPM) using non-linear classifier [50]. The readers are referred to [50] for more details on the LLC. It worth noting that according to our experimental results (see the next section), the codebook generated by K-Means can obtain satisfactory results. However in this work, the LLC coding criteria is used to train the codebook, which further improves the accuracy.

---

**Algorithm 1** LLC Incremental codebook optimization [50].

---

**input:**  $D_{init} \in \mathbb{S}^{128 \times M}$ ,  $S \in \mathbb{R}^{128 \times K}$ ,  $\mu$ ,  $\sigma$ ,    **output:**  $D_{opt}$

- 1:  $D_{opt} \leftarrow D_{init}$
- 2: **for**  $i = 1 : N$  **do**
- 3:     $d \leftarrow 1 \times M$  zero vector,
- 4:    **for**  $j = 1 : M$  **do**    // locality constraint parameters
- 5:      $d_j \leftarrow \exp^{-1}(-\|s_i - b_j\|^2/\sigma)$
- 6:    **end for**
- 7:     $d \leftarrow \text{normalize}_{(0,1)}(d)$
- 8:     $c_i \leftarrow \text{argmax}_c \|s_i - Dc\|^2 + \mu \|d \odot c\|^2$ ,    s.t.  $1^T c = 1$  // coding
- 9:     $id \leftarrow j \mid \text{abs}(c_i(j)) > 0.01$ ,  $D_i \leftarrow D(:, id)$  // remove bias
- 10:     $\tilde{c}_i \leftarrow \text{argmax}_c \|s_i - D_i c\|^2$ ,    s.t.  $\sum_j c(j) = 1$
- 11:     $\Delta D_i \leftarrow -2\tilde{c}_i(s_i - D_i \tilde{c}_i)$ ,     $\mu \leftarrow \sqrt{1/i}$  // update basis
- 12:     $D_i \leftarrow D_i - \mu \Delta D_i / \|\tilde{c}_i\|_2$
- 13:     $D_{opt}(:, id) \leftarrow \text{proj}(D_{D_i})$
- 14: **end for**

---

The final feature vectors of time-series signals are classified using a linear SVM classifier. It is worth noting that the resulting R-matrix by Formula (1) has only  $\{0,1\}$  values, that caused by thresholding parameter  $\epsilon$ . In order to avoid information loss by binarization of the R-matrix, this paper skips the thresholding step and uses the gray-level texture images. The proposed BoR pipeline for time-series classification is shown in Figure 4.

The proposed BoR model opens a new window of opportunities for time-series classification by treating the 1D signal classification as the computer vision problem of texture recognition. Note that the proposed pipeline is an exemplar case. Different features other than SIFT such as HOG [12] and LBP [37] can be extracted (or combination of these descriptors) in order to better represent the signals, that hopefully lead to a better accuracy of the over-all system.

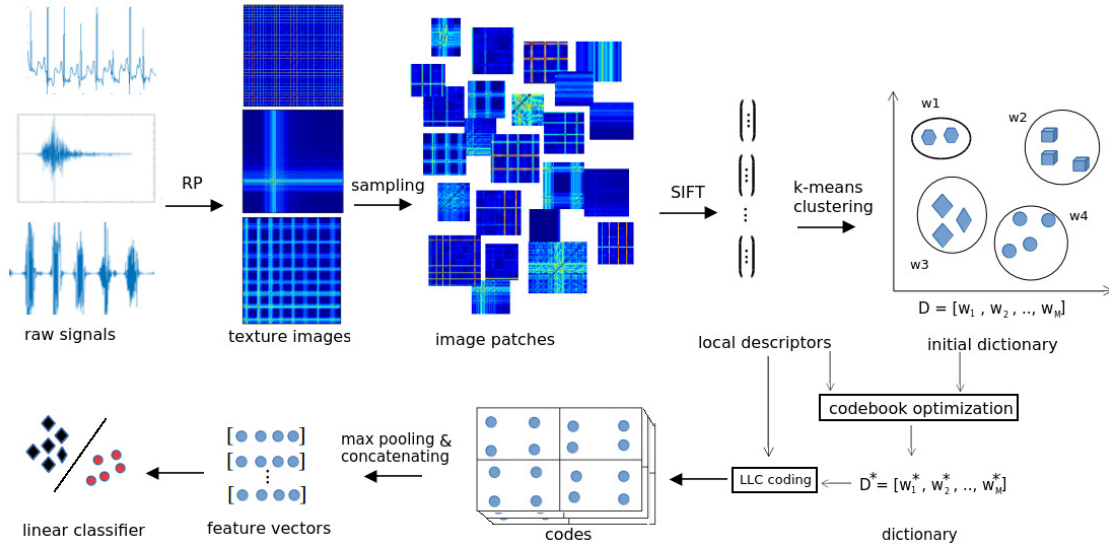


Fig. 4: The proposed BoR pipeline for time-series classification.

#### 4 Experimental results and discussions

In order to evaluate the performance of the proposed method, the UCR time-series classification archive is used [10]. For having a comprehensive evaluation of the classification algorithms, this repository contains 85 time-series datasets with different characteristics i.e. the number of classes  $2 \leq N_c \leq 60$ , number of training samples  $16 \leq N_{tr} \leq 8926$  and time-series length  $24 \leq l \leq 2709$ . The datasets are obtained from varieties of different real-world applications, ranging from EEG signals to food/beverage flavors, and electric devices to phonemes signals. We have selected the same 45 datasets that the results of the other state-the-art algorithms are also reported for them. The training and testing sets are separated to make sure that the results of different algorithms of different studies are comparable. Furthermore, the error rates of "1-NN Euclidean Distance", "1-NN DTW(r) where r is the percentage of time series length, and "1-NN DTW (no Warping Window)" are reported as the base-line methods. Only the results of 1-NN DTW is included in the table, because it obtains the lower recognition rate compared to the other two [10]. For a better comparison, 45 datasets that the results of the majority of related algorithms exist also for them are chosen (see Table 2). Application of RP (with the phase space dimension  $m = 3$ , and embedding time delay  $\tau = 4$ ) time-series to image encoding on first sample of five different datasets from the UCR archive are shown in Figure 5.

The obtained results are also compared to the state-of-the art BoF algorithms applied on time-series signals. One of the main work in this direction is time-series classification with a bag-of-features (TSBF) by [7]. Multiple ran-

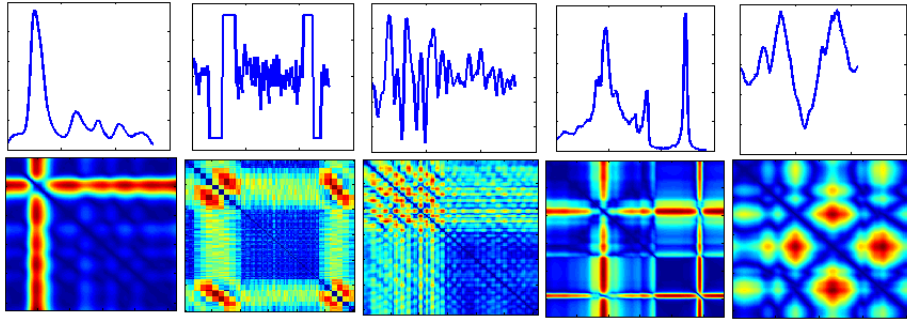


Fig. 5: Application of RP ( $m = 3, \tau = 4$ ) time-series to image encoding on five different datasets from the UCR archive: 50words, TwoPatterns, FaceAll, OliveOil and Yoga data (from left to right, respectively) without binarization.

| Parameters | patch size      | codebook size         | bag size | $C$ (SVM)            | $m$     | $\tau$   |
|------------|-----------------|-----------------------|----------|----------------------|---------|----------|
| Values     | { 16,24,32,48 } | { 50,100,250,...,8k } | >10,000  | $2^{-10}$ - $2^{10}$ | {3,4,5} | {1,3,10} |

Table 1: Different parameters of the proposed BoR model and their value ranges.

dom and uniform subsequences are sampled for feature extraction (e.g. slopes, means, and variances) and labeled with the class of the time-series. A random forest (RF) ensemble is used to both calculates "class probability estimates" (CPEs) for codebook and histogram of the CPEs to summarize the subsequence information. After adding the global features, a final classifier is used to assign each time series. Bag-of-Temporal-SIFT-Words (BoTSW) [5, 4] is the other algorithm that appears in the table. BoTSW tries to adapt SIFT image descriptors to time-series where the words correspond to the description of local gradients, computed at different scales, around keypoints.

For the comparison purposes, "Number of Wins" and "Average Rank" are calculated for each algorithm. Number of Wins counts number of datasets that a specific algorithm obtains the lowest error rates, while Average Rank is the mean of the error-rate ranking over entire datasets. Since some error rates are missing for some datasets for some algorithms, we normalized the measures for each algorithm. An algorithm outperforms the rest if it has a highest number of wins and lowest average rank.

For the computational efficiency, gray-level images with big dimensions resized down to maximum dimension of 300 pixels. In order to obtain image patches, a multi-scale Dense-SIFT method is used, with the different local image patch size, under 4 following scales:  $16 \times 16$ ,  $24 \times 24$ ,  $32 \times 32$  and  $48 \times 48$ . The optimum size of the codebook,  $M$ , is one of the most important parameters in BoF that has to be selected experimentally. Too small  $M$  has a limited discriminative ability, while a large size is likely to intro-

Table 2: Comparison of the proposed BoR method and the state-of-the art algorithms on the selected UCR datasets in terms of error rates (%).

| Dataset        | $N_c$ | $N_{tr}$ | $N_{te}$ | $l$  | 1-NN<br>DTW | SVM-<br>SAX | BoTSW[4] | BoF [7]     | proposed<br>BoR |
|----------------|-------|----------|----------|------|-------------|-------------|----------|-------------|-----------------|
| 50words        | 50    | 450      | 455      | 270  | 24.2        | 37.4        | 36.3     | <b>20.9</b> | 37.1            |
| Adiac          | 37    | 390      | 391      | 176  | 39.1        | 36.6        | 61.4     | 24.5        | <b>23.0</b>     |
| Beef           | 5     | 30       | 30       | 470  | 46.7        | 40.0        | 40.0     | 28.7        | <b>26.6</b>     |
| CBF            | 3     | 30       | 900      | 128  | <b>0.4</b>  | 4.8         | 5.8      | 0.9         | 1.8             |
| Chlori.        | 3     | 467      | 3840     | 166  | 35          | 34.2        | –        | 33.6        | <b>33.2</b>     |
| Cin.ECG        | 4     | 40       | 1380     | 1639 | <b>6.4</b>  | 10.4        | –        | 26.2        | 25.5            |
| Coffee         | 2     | 28       | 28       | 286  | 17.9        | 14.3        | <b>0</b> | 0.4         | <b>0</b>        |
| CricketX       | 12    | 390      | 390      | 300  | 23.6        | 42.6        | –        | 27.8        | <b>0</b>        |
| CricketY       | 12    | 390      | 390      | 300  | <b>19.7</b> | 34.6        | –        | 25.9        | 27.4            |
| CricketZ       | 12    | 390      | 390      | 300  | <b>17.9</b> | 39          | –        | 26.3        | 23.7            |
| Diatom.        | 4     | 16       | 306      | 345  | <b>6.5</b>  | 14.7        | –        | 12.6        | 9.8             |
| ECG200         | 2     | 100      | 100      | 96   | 12.0        | 18.0        | 11.0     | 14.5        | <b>10.8</b>     |
| ECG5day        | 2     | 23       | 861      | 136  | 20.3        | <b>14.0</b> | –        | 18.3        | 16.9            |
| FaceAll        | 14    | 560      | 1690     | 131  | 19.2        | 24.6        | 21.8     | 23.4        | <b>15.7</b>     |
| Face4          | 4     | 24       | 88       | 350  | 11.4        | <b>0</b>    | <b>0</b> | 5.1         | 9.0             |
| Fac.UCR        | 14    | 200      | 2050     | 131  | <b>8.8</b>  | 16.0        | –        | 9.0         | 21.7            |
| Fish           | 7     | 175      | 175      | 463  | 16.0        | 5.7         | 6.9      | 8.0         | <b>2.8</b>      |
| GunPoi.        | 2     | 50       | 150      | 150  | 8.7         | 2           | 8.0      | 1.1         | <b>0</b>        |
| Haptics        | 5     | 155      | 308      | 1092 | 58.8        | 56.2        | –        | 48.8        | <b>46.2</b>     |
| Inline.        | 7     | 100      | 550      | 1882 | <b>61.1</b> | 88.5        | –        | 60.3        | 61.4            |
| Ita.Pa.        | 2     | 67       | 1029     | 24   | <b>4.5</b>  | 9.2         | –        | 9.6         | 5.8             |
| Light.2        | 2     | 60       | 61       | 637  | <b>13.1</b> | 24.6        | 36.1     | 25.7        | 20.3            |
| Light.7        | 7     | 70       | 73       | 319  | 28.8        | 39.7        | 38.4     | 26.2        | <b>25.5</b>     |
| MALLAT         | 8     | 55       | 2345     | 1024 | 8.6         | 29.3        | –        | 3.7         | <b>3.4</b>      |
| Med.Im.        | 10    | 381      | 760      | 99   | <b>25.3</b> | 47.4        | –        | 26.9        | 34.8            |
| Mot.St.        | 2     | 20       | 1252     | 84   | <b>13.4</b> | 50.6        | –        | 13.5        | 19.0            |
| OliveO.        | 4     | 30       | 30       | 570  | 16.7        | 13.3        | 10.0     | <b>9.0</b>  | 20.8            |
| OSULea.        | 6     | 200      | 242      | 427  | 38.4        | 22.3        | 18.2     | 32.9        | <b>8.1</b>      |
| Son.Ro.        | 2     | 20       | 601      | 70   | 30.4        | 16.3        | –        | 17.5        | <b>16.0</b>     |
| So.RII         | 2     | 27       | 953      | 65   | <b>14.1</b> | 27.3        | –        | 19.6        | 26.2            |
| StarCu.        | 3     | 1000     | 8236     | 1024 | 9.5         | 4.2         | –        | <b>2.2</b>  | 3.9             |
| SwedLe.        | 15    | 500      | 625      | 128  | 15.5        | 14.4        | 15.2     | 7.5         | <b>6.8</b>      |
| Symbols        | 6     | 25       | 995      | 398  | 6.2         | 5.7         | –        | <b>3.4</b>  | 4.8             |
| SyntCo.        | 6     | 300      | 300      | 60   | 1.7         | 1.3         | 4.3      | 0.8         | <b>0.6</b>      |
| Thorax1        | 42    | 1800     | 1965     | 750  | 18.5        | –           | –        | 13.8        | <b>13.5</b>     |
| Thorax2        | 42    | 1800     | 1965     | 750  | 12.9        | –           | –        | 13.0        | <b>12.7</b>     |
| Trace          | 4     | 100      | 100      | 275  | 1.0         | <b>0</b>    | 1.0      | 2.0         | <b>0</b>        |
| TwoLead        | 2     | 23       | 1139     | 82   | 11.6        | 7.6         | –        | 4.6         | <b>3.1</b>      |
| TwoPat.        | 4     | 1000     | 4000     | 128  | 2.0         | 7.0         | 0.2      | 0.1         | <b>0</b>        |
| uWaveX         | 8     | 896      | 3582     | 315  | 22.7        | 76.5        | –        | <b>16.4</b> | 21.9            |
| uWaveY         | 8     | 896      | 3582     | 315  | 30.1        | 37.3        | –        | <b>24.9</b> | 29.5            |
| uWaveZ         | 8     | 896      | 3582     | 315  | 32.2        | 33.5        | –        | 21.7        | <b>20.2</b>     |
| Wafer          | 2     | 1000     | 6174     | 152  | 5.0         | 9.0         | 0.1      | 0.4         | <b>0</b>        |
| Wor.Syn.       | 25    | 267      | 368      | 270  | <b>25.2</b> | 45.8        | –        | 30.2        | 63.7            |
| Yoga           | 2     | 300      | 3000     | 426  | 15.5        | 16.1        | 15.0     | 14.9        | <b>13.7</b>     |
| <b># wins</b>  | –     | –        | –        | –    | 13/45       | 3/43        | 2/20     | 6/45        | <b>24/45</b>    |
| <b>AveRank</b> | –     | –        | –        | –    | 2.7         | 3.3         | 3.2      | 2.2         | <b>1.8</b>      |

duce noise. We used 5-fold cross-validation (CV) to select  $M$  from the set  $\{50, 100, 250, 500, 1000, 2000, 4000, 8000\}$ , with maximum number of k-means iterations set to 50. Regarding the LLC parameters, the number of neighbours was set to 5 with shift-invariant constraint. For assigning class labels, LIBSVM [9] is used as a linear classifier. The best  $C$  parameter of SVM was determined based on CV by varying  $C$  between 10 logarithmically equal spaced points between  $2^{-10}$  and  $2^{10}$ . Binary classifiers are extended by 1-vs-all method to address multi-class decisions (although techniques such as Error-Correcting Output Codes -ECOC- could have achieved a better performance [23,3,25,21] and we did not investigate it in this work).

Performances of the initial dictionary compared to the optimized one obtained by the LLC-based algorithm 1 for two arbitrarily chosen datasets are shown in Figure 6. It demonstrates the improvement of the input dictionary, regardless of the codebook size. However, the experiments show that the improvements are more significant for smaller size of codebooks and greater dictionary sizes benefit less from the optimization process.

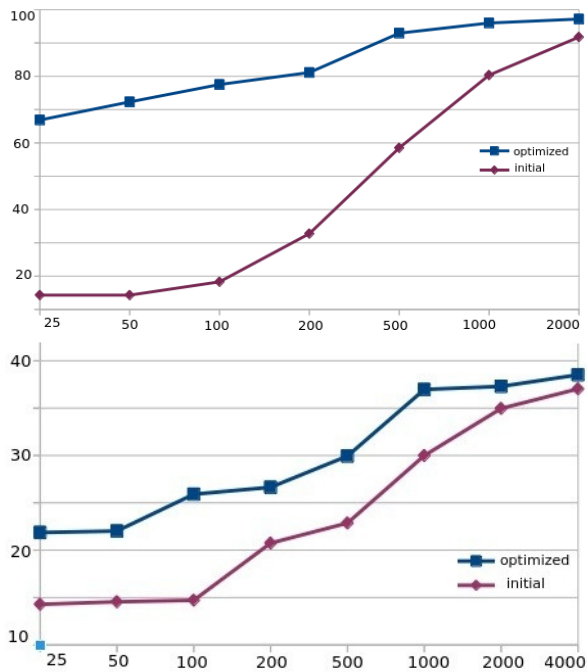


Fig. 6: Performance (in terms of recognition rates) of the initial dictionaries vs. optimized dictionaries over different codebook sizes for two arbitrarily chosen datasets (top:fish; bottom:inlineSkate datasets).

According to the Table 2, the proposed method obtained the lowest average ranking (1.8) and greater number of wins (24 out of 45). The second most

accurate algorithm, based on the average rank, is the BoF proposed by [7] that uses bag of signal segments in different scales and also takes advantage of the classifier ensemble (average rank = 2.2). However, if the number of wins is taken into account, 1NN DTW with 13 out of 45 is the third accurate algorithm. The results also demonstrate that although the 1NN DTW is a simple classifier, still achieves a better position compared to the popular SAX and BoTSW algorithms. The results also demonstrate that the transforming the time-series signal into 2D using RP in the BoF framework boosts the accuracy of the system.

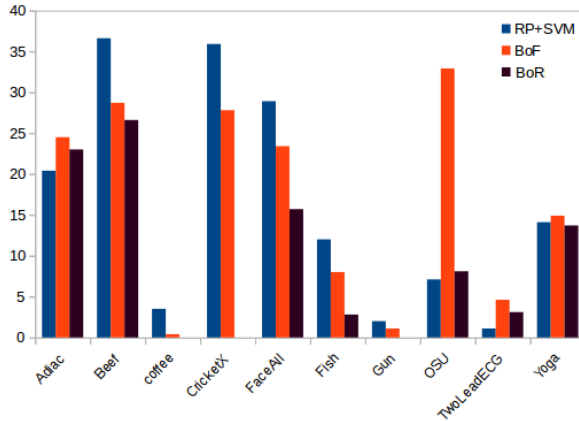


Fig. 7: Comparison (in terms of the error rate percentage) between BoF [7], RP+SVM [46] and their combination (BoR) on 10 datasets from the UCR archive.

It is known from the previous work that RP can boost time-series classification performance [46,20]. Furthermore, recently BoF classifiers widely and successfully applied for TSC tasks [7]. Figure 7 compares the performance of these two approaches against the proposed idea, which is somehow combination of them. Some experiments have been carried out in order to investigate which idea were responsible for most of the improvements, or is the combination itself that is uniquely useful. For 10 arbitrarily chosen datasets from the UCR archive, the proposed BoR obtains lower average error rate.

Figure 8 compares the performance of the proposed BoR model using different RP parameters e.g.  $m$  and  $\tau$ . Furthermore, we also investigate the impact of using gray-level images instead of the traditional binary images for TSC. The TwoLeadECG data from the UCR archive is chosen to run this experiment. Application of the RP on one sample using different parameters e.g.  $(m:1, \tau:5)$ ,  $(m:6, \tau:1)$ ,  $(m:6, \tau:4)$ ,  $(m:6, \tau:5)$ ,  $(m:3, \tau:4)$ ,  $(m:3, \tau:4, th:0.3)$ , and  $(m:3, \tau:4, th:0.5)$  is shown on the top of the Figure 8 from left to right, respectively. On the bottom, performances (in terms of accuracy) of the BoR model using different parameters is reported on the TwoLeadECG data. As visually

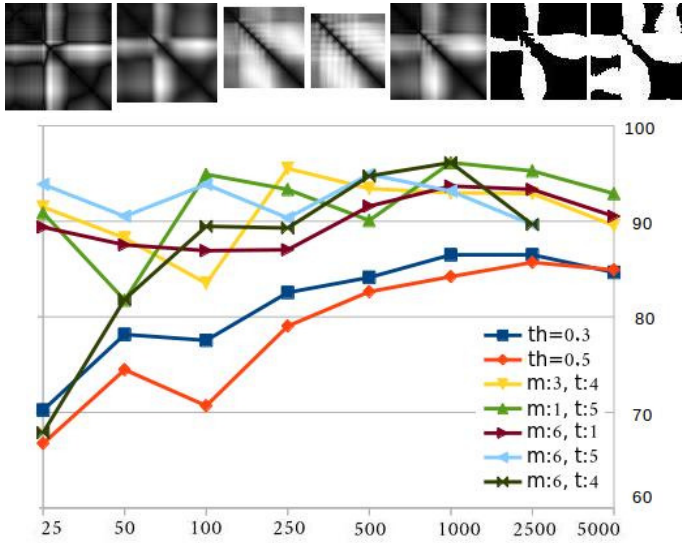


Fig. 8: Comparisons of different RP parameters on TwoLeadECG data of the UCR archive. Top: RP of first sample using (m:1,  $\tau$ :5), (m:6,  $\tau$ :1), (m:6,  $\tau$ :4), (m:6,  $\tau$ :5), (m:3,  $\tau$ :4), (m:3,  $\tau$ :4,  $th$ :0.3), and (m:3,  $\tau$ :4,  $th$ :0.5) from left to right, respectively; Bottom: performances (in terms of accuracy) of BoR model using different parameters.

noticeable, binarization of the RP images loses texture information (Fig. 8 top) and therefore, the lower performance of the BoR model for binary RPs compared to the gray-level images are explained (Fig. 8 bottom). Regarding the sensitivity analysis of the main parameters ( $m$ ,  $\tau$  and codebook size  $M$ ) of the BoR model, as shown, for a given ( $m$ ,  $\tau$ ) there is a suitable  $M$  that obtains a maximum performance ( $\sim 93\%$ , in this case).

Figure 9 compares the computational complexity (in terms of run-time) of the proposed BoR with the traditional 1D BoF model on the TwoLeadECG dataset. As shown, the LLC module of the proposed approach is consuming a significant time, in exchange for the accuracy boost (particularly for small  $M$ s, see Figure 6). However, the proposed BoR model without the LLC module has a similar run-time and also obtains a higher accuracy, compared to the traditional BoF model (usually requires a bigger  $M$  compared to the proposed BoR - see Figure 6).

## 5 Conclusions and future directions

This paper proposed image representation of time-series signals in BoF framework. By treating a time-series classification problem as a texture recognition task, this strategy facilitates application of new features from computer vision

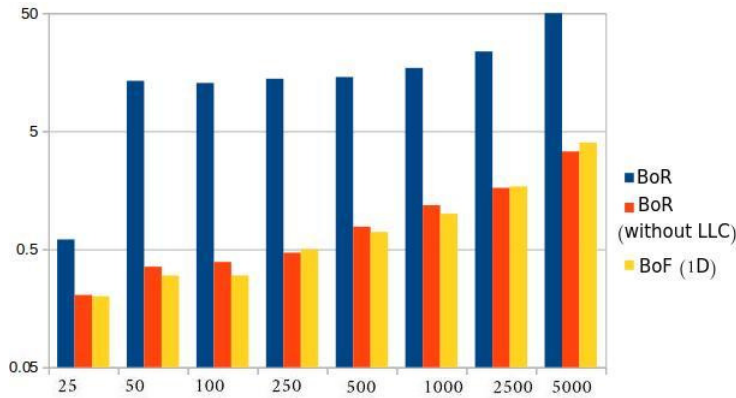


Fig. 9: Comparison of run-time (in seconds) of the proposed BoR with the traditional 1D BoF model on the TwoLeadECG dataset.

and image processing domains for time-series signals. This allows extracting more informative and discriminative information that can boost the recognition rate. The experimental results and analysis conclude that the proposed BoR approach has competitive performance when comparing to the 1D BoF applied on time-series. This suggests that for BoF approach, more discriminative and informative words can be obtained in 2D space compared to the words obtained from signal segments.

In this study, we used an existing time series-to-image transform i.e. RP for generating texture images. The future work will be devoted to finding/proposing a novel and more representative time-series to image transform. Furthermore, exploring different image descriptors such as LBP, gist, SURF and HoG, and also combination of them with SIFT may lead to a more discriminative dictionary that need to be investigated. Finally, performance of a hybrid method by combining words from both 1D segments and image representation of the words may also enrich the dictionary [22]. Application of the other classification algorithms such as deep neural networks is also worth investigating for boosting the recognition rate.

**Acknowledgements** This research is partially supported by the French national research agency (ANR) under the PANDORE grant with reference number ANR-14-CE28-0027.

N. Hatami also acknowledges support of the LABEX PRIMES (ANR-11-LABX-0063) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007).

## References

1. Acharya, U., Sree, S., Chattopadhyay, S., Yu, W., Ang, P., Application of recurrence quantification analysis for the automated identification of epileptic EEG signals, *International journal of neural systems* 21 (03), 199-211, 2011.



2. Allili, M., Wavelet modeling using finite mixtures of generalized Gaussian distributions: application to texture discrimination and retrieval, *IEEE Transactions on Image Processing* 21 (4), 1452-1464, 2012.
3. Armano, G., Chira, C., Hatami, N., 2012. Error-correcting output codes for multi-label text categorization. 3rd Italian Information Retrieval Workshop (IIR), 26-37.
4. Bailly, A., Malinowski, S., Tavenard, R., Guyet, T., Chapel, L., 2015. Bag-of-temporal-sift-words for time series classification. *ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data* .
5. Bailly, A., Malinowski, S., Tavenard, R., Guyet, T., Chapel, L., 2016. Dense bag-of-temporal-sift-words for time series classification. *Lecture Notes in Artificial Intelligence*.
6. Baydogan, M., Runger, G., 2016. Time series representation and similarity based on local auto patterns. *Data Mining and Knowledge Discovery* 30(2).
7. Baydogan, M., Runger, G., Tuv, E., 2013. A bag-of-features framework to classify time series. *IEEE Trans. on PAMI* 35(11), 2796-2802.
8. Bromuri, S., Zufferey, D., Hennebert, J., Schumacher, M., 2014. Multi-label classification of chronically ill patients with bag of words and supervised dimensionality reduction algorithms. *Journal of biomedical informatics* 51, 165-175.
9. Chang, C., Lin, C., 2011. Libsvm: a library for support vector machines. *ACM Trans. on Intelligent Systems and Technology* 2(3), 27.
10. Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., Batista, G., 2015. The ucr time series classification archive.
11. Chen, Y., Yang, H., Multiscale recurrence analysis of long-term nonlinear and nonstationary time series, *Chaos, Solitons & Fractals* 45 (7), 978-987, 2012.
12. Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. *CVPR, San Diego*.
13. Dong, Y., Feng, J., Liang, L., Zheng, L., Multiscale Sampling Based Texture Image Classification, *IEEE Signal Processing Letters*, Vol.24, Issue.5, 614-618, 2017.
14. Dong, Y., Tao, D., Li, X., Ma, J., Pu, J., Texture classification and retrieval using shearlets and linear regression, *IEEE transactions on cybernetics* 45 (3), 358-369, 2015.
15. Eads, D., Glocer, K., Perkins, S., Theiler, J., 2005. Grammar-guided feature extraction for time series classification. *NIPS*.
16. Eckmann, JP. Kamphorst, SO. Ruelle, D., Recurrence Plots of Dynamical Systems, *Europhysics Letters*, 4(9), 973-977. (1987).
17. Fu, Z., Lu, G., Ting, K., Zhang, D., 2011. Music classification via the bag-of-features approach. *Pattern Recognition Letters* 32(14), 1768-1777.
18. Geurts, P., 2009. Pattern extraction for time series classification. 5th EU Conf. Principles of Data Mining and Knowledge Discovery, 115-127.
19. Grabocka, J., Wistuba, M., Schmidt-Thieme, L., 2015. Scalable classification of repetitive time series through frequencies of local polynomials. *IEEE Trans. on Knowledge and Data Engineering* 27(6), 1683-1695.
20. Hatami, N., Gavet, Y., Debayle, J., Classification of Time-Series Images Using Deep Convolutional Neural Networks, *International Conference on Machine Vision (ICMV)*, 2017.
21. Hatami, N., 2008. Thinned ecoc decomposition for gene expression based cancer classification. 8th IEEE Conf. on Intelligent Systems Design and Applications, 216-221.
22. Hatami, N., 2012a. Some proposals for combining ensemble classifiers. PhD thesis, University of Cagliari.
23. Hatami, N., 2012b. Thinned-ecoc ensemble based on sequential code shrinking. *Expert Systems with Applications* 39(1), 936-947.
24. Hatami, N., Chira, C., 2013. Classifiers with a reject option for early time-series classification. *IEEE Sym. on Computational Intelligence and Ensemble Learning (CIEL)*, 9-16.
25. Hatami, N., Ebrahimpour, R., Ghaderi, R., 2008. Ecoc-based training of neural networks for face recognition. *IEEE Conf. on Cybernetics and Intelligent Systems*, 450-454.
26. Jeong, Y., Jeong, M., Omitaomu, O., 2011. Weighted dynamic time warping for time series classification. *Pattern Recognition* 44(9), 2231-2240.
27. Keogh, E., Kasetty, S., 2003. On the need for time series data mining benchmarks: A survey and empirical demonstration. *Data Mining and Knowledge Discovery* 7(4), 349-371.

28. N. Kumar, V. Lolla, E. Keogh, S. Lonardi, C. Ratanamahatana, L. Wei, 2005. Time-series bitmaps: a practical visualization tool for working with large time series databases, SIAM international conference on data mining, 531-535.
29. Lategahn, H., Gross, S., Stehle, T., Aach, T., Texture classification by modeling joint distributions of local patterns with Gaussian mixtures, IEEE Transactions on Image Processing 19 (6), 1548-1557, 2010.
30. Lin, J., Khade, R., Li, Y., 2012. Rotation-invariant similarity in time series using bag-of-patterns representation. Journal of Intelligent Information Systems 39(2), 287-315.
31. Lowe, D., 2004. Distinctive image features from scale-invariant keypoints. International journal of computer vision 60(2), 91-110.
32. Marwan, N., Romano, M.C., Thiel, M., Kurths, J., 2007. Recurrence plots for the analysis of complex systems. Physics Reports 438(5-6), 237-329.
33. Marwan, N., Wessel, N., Meyerfeldt, U., Schirdewan, A., Kurths, J. (2002). Recurrence Plot Based Measures of Complexity and its Application to Heart Rate Variability Data. Physical Review E. 66 (2): 026702.
34. Marwan, N. Kurths, J. Comment on stochastic analysis of recurrence plots with applications to the detection of deterministic signals by rohde et al.[physica d 237 (2008) 619629], Physica D 238 (2009) 1711-1715.
35. Mehta, R., Eguiazarian, K., Texture classification using dense micro-block difference, IEEE Transactions on Image Processing 25 (4), 1604-1616, 2016.
36. Nanopoulos, A., Alcock, R., Manolopoulos, Y., 2001. Feature-based classification of time-series data. Int. J. Computer Research 10, 49-61.
37. Ojala, T., Pietikainen, M., Maenpaa, T., 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. TPAMI 7(24), 971-987.
38. Ratanamahatana, C., Keogh, E., 2004. Making time-series classification more accurate using learned constraints. Proc. SIAM Intl Conf. Data Mining, 11-22.
39. Rodriguez, J., Alonso, C., 2004. Interval and dynamic time warping-based decision trees. ACM Symp. Applied Computing, 548-552.
40. Rodriguez, J., Alonso, C., Maestro, J., 2005. Support vector machines of interval-based features for time series classification. Knowledge-Based Systems 18, 171-178.
41. Rohde, G., Nichols, J., Dissinger, B., Bucholtz, F., Stochastic analysis of recurrence plots with applications to the detection of deterministic signals, Physica D: Nonlinear Phenomena, 237(5), 619-629, 2008.
42. Roma, G., Nogueira, W., Herrera, P., de Boronat, R., Recurrence quantification analysis features for auditory scene classification, IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events, 2013.
43. Schafer, P., 2015. The boss is concerned with time series classification in the presence of noise. Data Mining and Knowledge Discovery 29(6), 1505- 1530.
44. Senin, P., Malinchik, S., 2013. Sax-vsm: Interpretable time series classification using sax and vector space model. Data Mining (ICDM), IEEE 13th Int. Conf. on, 1175-1180.
45. Souza, V. Silva, D. Batista, G., Time Series Classification Using Compression Distance of Recurrence Plots, IEEE 13th International Conference on Data Mining, 687-696, 2013.
46. Souza, V. Silva, D. Batista, G., 2014. Extracting texture features for time series classification, Pattern Recognition (ICPR), 22nd International Conference on, 1425-1430.
47. Thiel, M. Romano, MC. Kurths, J. Analytical description of recurrence plots of white noise and chaotic processes, arXiv:nlin/0301027, 2003.
48. Ueno, K., Xi, X., Keogh, E., Lee, D., 2007. Anytime classification using the nearest neighbour algorithm with applications to stream mining. IEEE IntlConf. Data Mining, 623-632.
49. Wang, J., Liu, P., She, M., Nahavandi, S., Kouzani, A., 2013. Bag-of-words representation for biomedical time series classification. Biomedical Signal Processing and Control 8(6), 634-644.
50. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y., 2010. Locality-constrained linear coding for image classification. CVPR, 3360-3367.
51. Wang, Z., Oates, T., 2014. Time warping symbolic aggregation approximation with bag-of-patterns representation for time series classification. Machine Learning and Applications (ICMLA), 13th Int. Conf. on, 270-275.
52. Wang, Z., Oates, T., 2015. Pooling sax-bop approaches with boosting to classify multivariate synchronous physiological time series data. FLAIRS Conference, 335-341.

53. Xing, Z., Pei, J., Yu, P., 2011. Early prediction on time series: A nearest neighbor approach. *Int. Joint Conf. Artificial Intelligence* 2168, 1297-1302.
54. Yang, J., Yu, K., Gong, Y., Huang, T., 2009. Linear spatial pyramid matching using sparse coding for image classification. *CVPR*, 1794-1801.
55. Yang, H., Multiscale recurrence quantification analysis of spatial cardiac vectorcardiogram signals, *IEEE Transactions on Biomedical Engineering*, Vol.58, No.2, 339-347, 2011.
56. Zbilut, J.P., Webber Jr., C.L. (1992). Embeddings and delays as derived from quantification of recurrence plots. *Physics Letters A*. 171 (3-4): 199-203.
57. Zhang, M., Sawchuk, A., 2012. Motion primitive-based human activity recognition using a bag-of-features approach. *ACM SIGHIT sym. on Int. health informatics (IHI)*, 631-640.
58. Zhao, J., Itti, L., 2016. Classifying time series using local descriptors with hybrid sampling. *IEEE Trans. on Knowledge and Data Engineering* 28, 623-637.