Blind source separation and feature extraction in concurrent control charts pattern recognition: Novel analyses and a comparison of different methods

Guilherme Dean Pelegrina, Leonardo Tomazeli Duarte, Christian Jutten

Blind source separation and feature extraction in concurrent control charts pattern recognition: Novel analyses and a comparison of different methods

Guilherme Dean Pelegrina[a], Leonardo Tomazeli Duarte[a], Christian Jutten[b]

[a]*School of Applied Sciences, University of Campinas (UNICAMP), Limeira, Brazil*
[b]*GIPSA-lab, Université Joseph Fourier, Grenoble, France*

Abstract

Control charts are among the main tools in statistical process control (SPC) and have been extensively used for monitoring industrial processes. Currently, besides the single control charts, there is an interest in the concurrent ones. These graphics are characterized by the simultaneous presence of two or more single control charts. As a consequence, the individual patterns may be mixed, hindering the identification of a non-random pattern acting in the process; this phenomenon is refered as concurrent charts. In view of this problem, our first goal is to investigate the importance of an efficient separation step for pattern recognition. Then, we compare the efficiency of different Blind Source Separation (BSS) methods in the task of unmixing concurrent control charts. Furthermore, these BSS methods are combined with shape and statistical features in order to verify the performance of each one in pattern classification. In additional, the robustness of the better approach is tested in scenarios where there are different non-randomness levels and in cases with imbalanced dataset provided to the classifier. After simulating different patterns and applying several separation methods, the results have

shown that the recognition rate is widely influenced by the separation and feature extraction steps and that the selection of efficient separation methods is fundamental to achieve high classification rates.

1. Introduction

Given the competitiveness between companies in order to produce with minimal defects, quality control is a fundamental issue in the industrial environment. In this context, a set of tools, known as Statistical Process Control (SPC), is useful for monitoring the stability and capability of the process under analysis, detecting potential failures and minimizing the number of defects [1, 2]. The most widespread tool of SPC is the control chart, proposed by Shewhart in 1924 [1]. Based on a set of process parameters, this tool verifies if the variability observed in a process stems from natural or specific causes. The latter case is characterized by points that are outside the control limits (Figure 1, dashed horizontal lines) or by unnatural (or abnormal) patterns, indicating that the process is statistically out of control [1, 2].

Unnatural patterns are associated with assignable causes which perturb the expected behavior of the process [3]. Western Electric Company [3] shows a set of assignable causes for several patterns, which includes stratification, systematic, cyclic, trend and shift. In Figure 2, we show examples of abnormal patterns as well as an example of a normal one observed in the case of natural causes of variability. Traditionally, the identification of these pat-

2
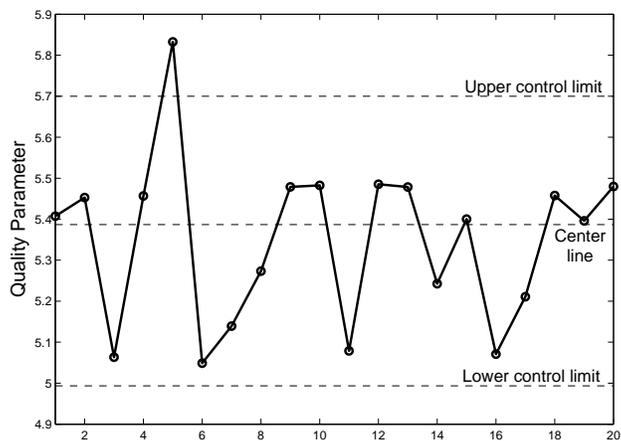
Figure 1: Example of a control chart.

terns are based on appropriate rules (e. g. running rules) [3, 4]. However, this procedure requires trained experts and may lead to false alarms. In view of this problem, there is a need to develop intelligent systems methods to recognize unnatural control chart patterns and, then, to detect assignable causes prematurely.

Several approaches have been developed to automatically identify unnatural patterns [5, 6, 7, 8, 9, 10, 11]. A crucial step of these methods is feature extraction, which provides the discriminant parameters used to classify the different patterns. Existing approaches consider the wavelet analysis [5, 6] and statistical features (autocorrelation coefficient, regression [7], mean, standard deviation, skewness and kurtosis [8]) as inputs of the classifier. Besides these features, a great number of methods are based on shape features [7, 9, 10, 11], and its results have shown that the use of shape features as inputs of the classifier discriminates well the patterns, leading to

3
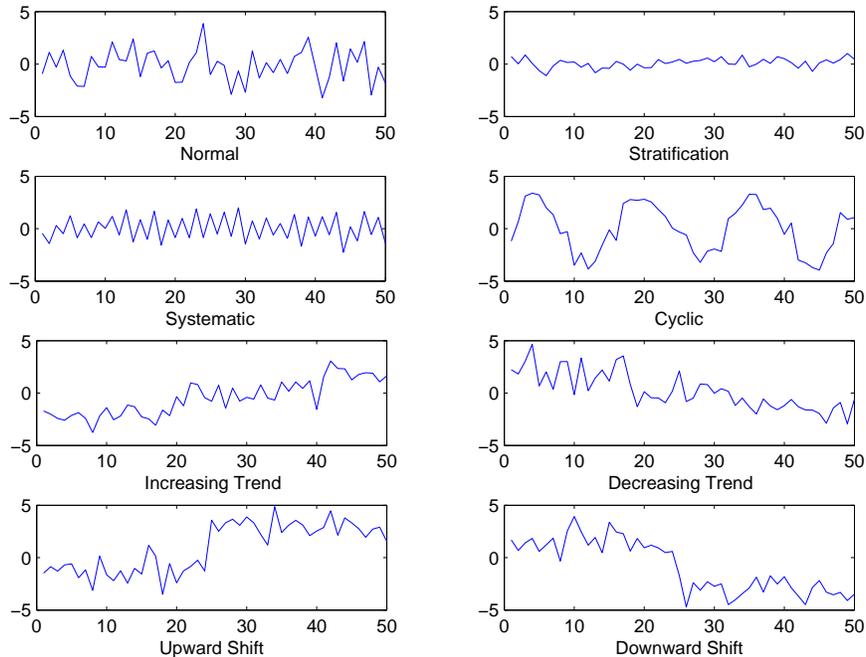
Figure 2: Examples of observed patterns in control charts.

higher classification rates. Ranaee, Ebrahimzadeh and Ghaderi [8], Zhang and Cheng [12] and Wu, Liu and Zhu15 [13] developed a method composed of statistical and shape features as inputs of a support vector machine (SVM) classifier. In [12], genetic algorithm was applied as an optimization tool to improve the performance of the multiclass SVM classifier. The model presented in [13] uses shape and statistical features as inputs of a binary-class SVM which identifies if the pattern is normal or not and, if the latter case is true, a binary-tree SVM is employed to recognize the pattern involved. Xanthopoulos and Razzaghi [14] also utilized the SVM classifier, but a weighted apporach. The results have shown the benefits when the data are highly

4

imbalanced.

These latter works consider the recognition based on single control charts. However, the data observed may be a mixture of two or more control charts acting simultaneously in the process. This graphics, called concurrent control charts, are commonly present in machining process. For example, the tool may wear in turning process (resulting in a trend pattern) and this can coexist with periodic reposition of unstable material [15, 16] or voltage fluctuation [17] (resulting in a cycle pattern). As a consequence, the individual patterns may be mixed and their characteristics lost, making the detection of an abnormality difficult. Figure 3 shows two control charts with different patterns (normal and systematic) that were mixed. One can visually note the difficulty in identifying the two patterns clearly.

In order to deal with the issue of concurrent patterns, a class of methods addresses the problem by directly processing the mixed charts. For instance, Guh and Tannock [15] developed a classifier based on a backpropagation network and considered two situations: a first one comprising only non-random patterns and a second one where it arises progressively. The results have shown that the approach suits well the first case while being limited to deal with the latter one. Yang and Yang [18] used a statistical correlation coefficient in order to recognize unnatural patterns in both single and concurrent control charts.

Another strategies aim at separating the individual characteristics at first. This can be done, for instance, by considering the wavelet transform to decompose the concurrent pattern into two basic patterns in onder to be easily recognized by a neural network model [17] or a multiclass support vector
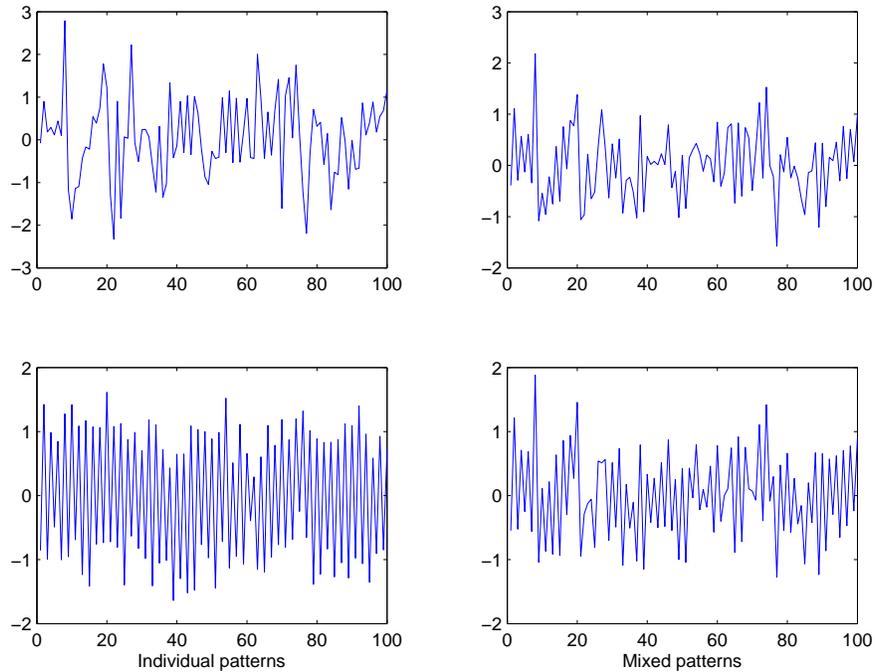
Figure 3: Example of patterns that was mixed.

machines [19]. Yang et al. [16] proposed a hybric method that integrates extreme-point symetric mode decomposition with extreme learning machine. Such an approach allows one to identify concurrent charts but also to estimate which pattern provides the major contribution into the observed mixture. Gu et al. [20] and Xie et al. [21] applied singular spectrum analysis to decompose the concurrent patterns utilized learn vector quantization [20] and support vector machine [21] to recover the individual patterns.

Some other approaches have applied the independent component analysis (ICA), a methodology that was developed to solve the problem of blind source separation (BSS) [22, 23], to extract the independent components

6

(ICs) from the concurrent control charts. Wang, Dong and Kuo [24] proposed a hybrid approach integrating ICA and decision tree. Considering shape and statistical features as inputs of the classifier, results have indicated that such approach suits well the problem in most cases involving concurrent control chart patterns. Lu, Shao and Li [25] adopted a SVM-based classifier and considered the cases of raw data, shape and statistical features (without ICA) and independent components. The results have also shown the robustness of the ICA-SVM approach in detecting unnatural patterns in concurrent control charts.

As can be seen in the literature, the shape and statistical features are widely used in the single control charts patterns recognition and ICA is an efficient method to separate the original data providing satisfactory classification rates of the concurrent case. In this context, this paper proposes four main analyses. Initially, we verify how the feature extraction step, including the separation, is important in the classification. In addition, the efficiency of several BSS methods, that are not necessarily based on ICA, are compared in the separation of concurrent control charts. Then, we perform the classification based on the data provided by the BSS methods and on the shape and statistical parameters extracted from these data. Finally, after selecting the configuration that provides the better classification rates, we assess its performance for different non-randomness levels and imbalanced dataset used in the training step of the classifier.

The rest of this paper is organized as follows. Section 2 describes the methodological aspects related to our analyses. Section 3 presents the experiments and the results obtained are discussed. In Section 4 we provide our

conclusions on this study.

2. Methodology

As described in Figure 4, a general approach for automatic classification of concurrent control charts comprises three steps. The first one is related to BSS and can be seen as a data pre-processing step whose goal is to separate the individual control charts, i. e. to provide an estimated dataset (BSS-data) from the observed mixtures (raw data). Then, in the second step, a feature extraction step aims at extracting relevant information from the BSS-data by taking into account either statistical or shape features. Finally, the data provided by feature extraction (processed data) feed a pattern recognition method, which estimates the original categories of the individual control charts.

An illustrative example, as mentioned previously, is shown as follows. If we consider the tool wear in a turning process (unknown trend pattern) and a voltage fluctuation (unknown cycle pattern), the observed data corresponds to mixtures of the single patterns involved. Applying a BSS-method, we are able to recover the control charts containing the trend and cyclic patterns. In possession of the single data, we extract the shape and statistical features which are used as input of the classifier. Based on the training structure, the classifier provides the recognition of the control charts pattern involved in the process, in this case, trend and cyclic patterns.

In this section, we present the theoretical aspects that compose the structure shown in Figure 4. First of all, we provide a brief introduction to the BSS methods considered in this work. Then, we discuss the main shape fea-

Figure 4: Classification structure.
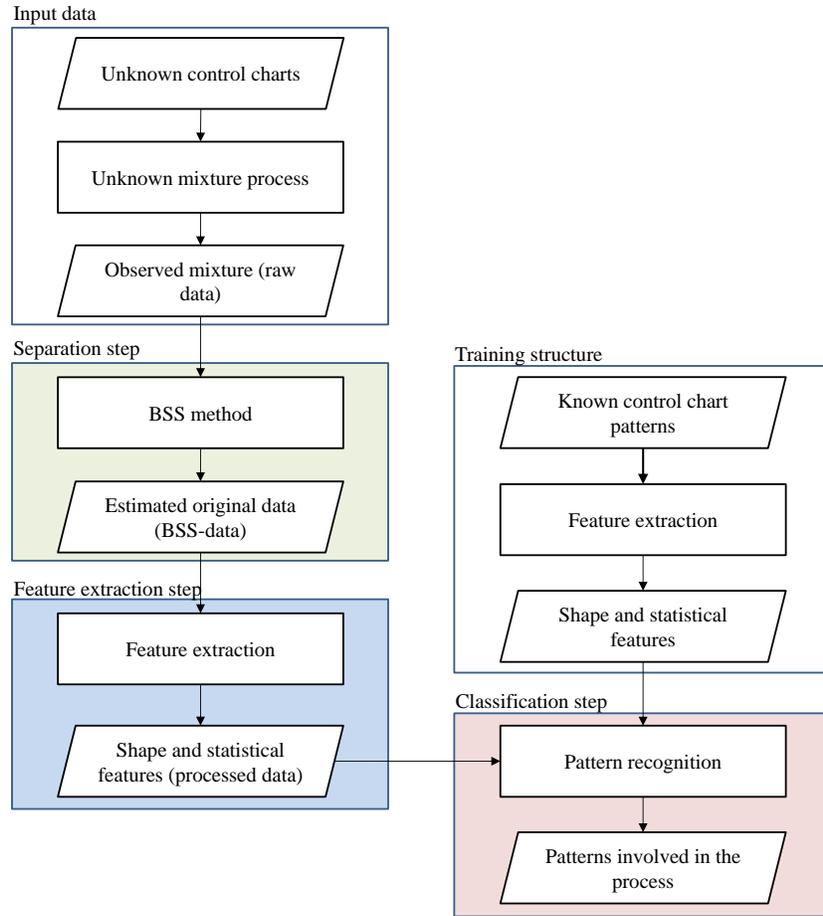
tures that are usually adopted in the context of control charts. Finally, the pattern recognition step is described.

2.1. Blind Source Separation problem

In summary, the BSS problem consists in recovering a set of signal sources from a set of mixed data of these sources, without the knowledge of both the original signals and the mixing process [23]. In this paper, we consider

the most common formulation of the BSS problem, in which the number of sources (single control charts) to be recovered is equal to the number of mixed observed data (concurrent control charts). Thus, assuming that the mixing process is linear and instantaneous, for each sample t ($t = 1, \ldots, T$), the sources can be represented by the vector $\mathbf{s}(t) = [s_1(t) \ s_2(t) \ \cdots \ s_n(t)]^T$ and the mixed signals by the vector

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) \ , t = 1, \ldots, T \tag{1}$$

where $\mathbf{A}$ corresponds to a square matrix $n \times n$ representing the mixing process.

Assuming that the sources $s_i(t)$, $i = 1, \ldots, n$, are statistically mutually independent, it is easy to show that observations $x_j(t)$, $j = 1, \ldots, n$ as mixtures of the sources, are no longer independent. The central idea of many source separation methods is to estimate a separating system, modeled by a matrix $\mathbf{B}$, so that

$$\mathbf{y}(t) = \mathbf{B}\mathbf{x}(t) \tag{2}$$

has the independence property [26]. In fact, following Darmois's results [27] and Comon's paper [28], one can prove that the above problem has no solution if the sources (i.e. the temporal patterns, in our application) have samples which are temporally independent and identically distributed (i.i.d.) and Gaussian. Consequently, there are two ways for solving the problem. Assuming that sources are i.i.d. and non-Gaussian leads to ICA which requires higher (than two) order statistics (HOS). Assuming that sources are non i.i.d. and (possibly) Gaussian, source separation can be achieved using only second order statistics (SOS), i.e. through simpler and faster algorithms.

10

Practically, having non i.i.d. source means that either the source is temporally colored (i.e. successive samples are not independent) or the source is non-stationary (i.e. successive samples are not identically distributed).

In the context of control chart, the successive samples related to any pattern are colored, with correlation functions which vary according to the patterns. Consequently, despite the fact that previous works that apply BSS to concurrent control charts only consider HOS-based source separation methods, SOS-based source separation methods should be suited to our problem as well.

2.1.1. Higher-Order Statistics methods

In concurrent control chart pattern recognition, previous works have mainly focused on the ICA framework [24, 25], which can be seen as a HOS method. In this study, we consider two distinct HOS-based approaches to estimate the separation matrix: maximization of non-Gaussianity and maximum likelihood [22, 23].

In methods based on the maximization of non-Gaussianity, such as FastICA [22], the separating matrix $\mathbf{B}$ is adjusted to maximize a measure of non-Gaussianity. In this approach, it is necessary to perform a process known as whitening, which aims at decorrelating the mixtures [22, 23]. After whitening, with the whitening matrix W, one obtains:

$$\mathbf{z}(t) = \mathbf{WAs}(t) \tag{3}$$

where $E\{\mathbf{z}(t)\mathbf{z}^T(t)\} = \mathbf{I}$. Moreover, it is easy to show that the matrix $\mathbf{WA}$ is orthogonal [22]. Thus, the cancellation of the mixing matrix $\mathbf{WA}$ is obtained by the estimation of an orthogonal separating matrix. Consequently, after

11

whitening, the parametric model of separating matrix is simpler.

Two measures are usually adopted to quantify non-Gaussianity: kurtosis and negentropy [22, 23]. The kurtosis of a random variable y, defined by

$$kurt(y) = E\{y^4\} - 3(E\{y^2\})^2 \ ,$$

(4)

corresponds to the fourth-order moment of y. High values of kurtosis (in modulus) indicates a dataset that is far from Gaussian [22, 23].

The other strategy used to measure non-Gaussianity, the negentropy, is based on the concept of entropy from information theory. Since the Gaussian distribution has the highest entropy measure among all continuous random variables with the same variance [22], this information can be used to formulate a criterion of non-gaussianity. Therefore, the negentropy can be defined by the following expression:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \ ,$$

(5)

where $\mathbf{y}_{gauss}$ is a Gaussian random vector whose correlation matrix is the same that of $\mathbf{y}$ and $H(\mathbf{y})$, defined by

$$H(\mathbf{y}) = -\int p_y(\eta) \ \log p_y(\eta) \ d\eta \ ,$$

(6)

is the entropy of the random vector $\mathbf{y}$, whose probability density is $p_y(\eta)$.

The second approach of HOS-based source separation is to use maximum likelihood for estimating $\mathbf{B}$ from the distribution of the observed data $\mathbf{x}(t)$ [29, 30]. Given the independence assumption of the sources $\mathbf{s}(t)$ and the observation model $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$, the likelihood of observation $\mathbf{x}(t)$, given $\mathbf{B}$, can be written as:

$$p_x(\mathbf{x}|\mathbf{B}) = |\det \mathbf{B}| \, p_s(\mathbf{s}) = |\det \mathbf{B}| \prod_i p_{s_i}(s_i) \ ,$$

(7)

where $\mathbf{B} = \mathbf{A}^{-1}$ and $p_{s_i}(s_i)$ is the probability density function of each independent component. Given the set of observations, the maximum likelihood approach in this case aims to determine the matrix $\mathbf{B}$ that maximizes (7).

The Infomax approach to BSS was proposed by Bell and Sejnowski [31] and, as demonstrated by Cardoso [19], it is close-related to the maximum likelihood approach, thus leading to similar practical algorithms.

2.1.2. Second-Order Statistics methods

Methods based on SOS have been applied in different areas [32, 33]. Differently from BSS methods based on HOS, as we explained above, SOS techniques exploit the temporal correlation of the observed data to find the separation matrix, thus allowing the separation of sources that have a temporal structure. In other words, in SOS methods, the sources are not modeled as i.i.d. (independent and identically distributed) processes [23]. An interesting feature of SOS methods is the capability to separate non i.i.d. Gaussian signals, which are common in control charts and cannot be separated by HOS methods - it is worth noticing that previous contributions on control charts classification did not exploit SOS methods.

A first way to implement SOS methods is to search for a separating matrix so that two or more covariance matrices of the retrieved signals for different delays τ, denoted by $\mathbf{R_y}(\tau) = E[\mathbf{y}(t)\mathbf{y}^T(t-\tau)]$, be diagonal matrices. A first line of methods considers the exact joint diagonalization of two correlation matrices, which led to an algorithm known as AMUSE [34].

More generally, the techniques based on approximate joint diagonalization of more than two matrices, like SOBI [35] and WASOBI [36], are currently more used due the higher accuracy in separation. The goal is to find a

unique matrix $\mathbf{U}$ which leads to a set of diagonal covariance matrices $\mathbf{R_y}(l)$ with different delays l, $l = 0, 1, \ldots, L$. This procedure is accomplished by solving for $\mathbf{U}$ the set of equations $\mathbf{R_z}(l) = \mathbf{U}\mathbf{D}_l\mathbf{U}^T$, $l = 0, 1, \ldots, L$, where $\mathbf{D}_l$ are diagonal matrices and $\mathbf{R_y}(l)$ corresponds to the covariance matrix of the whitened data [23].

In mathematical terms, diagonalization can be formulated by zeroing the off-diagonal entries. A simple criterion for that is:

$$off[\mathbf{R_y}(l)] = \sum_{1 \leq i \neq j \leq n} |R_{\mathbf{y}_{ij}}(l)|^2. \tag{8}$$

The unitary diagonalization of $\mathbf{R_z} = [\mathbf{R_z}(1), \ldots, \mathbf{R_z}(L)]$ is obtained when $off(\mathbf{V}^T\mathbf{R_z}\mathbf{V})$ is set to zero by adjusting a unitary matrix $\mathbf{V}$. If $\mathbf{R_z} = \mathbf{U}\mathbf{D}\mathbf{U}^T$, where $\mathbf{U}$ is unitary and $\mathbf{D}$ is diagonal, $\mathbf{V}$ is essentially equal to $\mathbf{U}$. The criterion therefore is to minimize

$$C(\mathbf{R_z}, \mathbf{V}) = \sum_{l=1,\ldots,L} off[\mathbf{V}^T\mathbf{R_z}(l)\mathbf{V}], \tag{9}$$

After solving (9), one obtains the unitary matrix $\mathbf{U}$. Furthermore, from the relation $\mathbf{A} = \mathbf{W}^{-1}\mathbf{U}$, where $\mathbf{W}$ is the whitening matrix, we estimate the mixing matrix $\mathbf{A}$ [35].

2.2. Shape and statistical features selection

In order to assess the influence of the feature extraction step, including separation and shape/statistical parameters, we consider two different inputs in this study. The first is simply to use all the samples in the vector $\mathbf{y}$ as the input of the classifier, i.e., the BSS-data extracted from the BSS method. The second approach takes as input some shape features and also statistical

features extracted from $\mathbf{y}$ (processed data). Based on the literature [7, 8, 9, 10, 11, 12, 13], the shape features considered in this study are:

- Number of crossovers of the pattern i with its mean line ($CPML_i$):

$$CPML_i = \sum_{t=1}^{T-1} c_t, \quad i = 1, 2, \ldots, n, \tag{10}$$

where $c_t = 1$ if $(y_i(t) - \overline{y}_i)(y_i(t+1) - \overline{y}_i) < 0$ or $c_t = 0$ otherwise, $y_i(t)$ is the sample t in the control chart i for each time point $t = 1, 2, \ldots, T$ and $\overline{y}_i$ is the mean value. This feature can discriminate systematic patterns from the others, once its value is higher. For normal pattern, the $CPML_i$ value is intermediate.

- Average value of the segment slopes (SS_i):

$$SS_i = \frac{\sum_{k=1}^{K} q_{i,k}}{K}, \quad i = 1, 2, \ldots, n, \tag{11}$$

where $q_{i,k}$ is the slope value of the segment k, $k = 1, 2, \ldots, K$ and K is the number of segments (in this study, $L = 4$). By dividing the pattern into equal segments and calculate the slopes of the least squares lines for each one, we can discriminate trend patterns from the others, whose values are higher. Normal and systematic patterns have values close to zero.

- Average value of the difference between the slopes of the pattern least square line and of the segments ($DSSP_i$):

$$DSSP_i = \frac{\sum_{k=1}^{K} q_{i,k} - p_i}{K}, \quad i = 1, 2, \ldots, n, \tag{12}$$

where q_k is the slope value of the segment k, $k = 1, 2, \ldots, K$, K is the number of segments and p_i is the slope of the pattern. These values for upward and downward shifts are high, so this feature is important to discriminate the latter patterns from the others ones. Normal and systematic patterns have also values close to zero.

- Maximum value of the autocovariance (MAC_i):

$$MAC_i = max(E[(y_i(t))(y_i(t+m))^T]), \ m = T/3, \ldots, 2T/3, \ i = 1, 2, \ldots, n,$$

(13)

where $E[(y_i(t))(y_i(t+m)^T]$ is the autocovariance considering m delays and T is the number of samples in the control chart i. With the delays varying from $m = T/3$ to $m = 2T/3$ ($m = 33$ to $m = 67$, considering, in our case, $T = 100$), we can discriminate well systematic and cyclical patterns, once its autocovariance for these delays is higher comparing to the others.

2.3. Pattern recognition

In this step, the information obtained from the separation methods and/or the shape/statistical features extraction are used by the classifier to determine which patterns are involved in the process. This study considers offline classification and our analysis is based on Support Vector Machine (SVM) classifiers [37]. This classifier has been widely applied to process control [8, 12, 13, 14, 25]. Briefly, the SVM is a classifier that adjust an optimal hyperplane that provides the separation of the classes with the largest margin. As a supervised method, the SVM has two steps to follow: training and testing. In the first step we need to "teach" the algorithm that a particular

16

set of data belongs to a specific class. In the second step, the data obtained from the BSS method and the shape/statistical feature extraction are used as the input of the SVM and the results are obtained. It is worth noting that, since we compare the results using BSS-data after the BSS method and processed data, the set of data used in training are depending of the classifier input.

3. Experiments and results

In order to better exploit the application of BSS techniques to concurrent control charts, the experiments are divided in four parts. The control charts considered in each experiment were obtained from an automatic pattern generator based on the literature [7, 9, 10, 11, 13, 17, 18, 21, 24, 25] and described in Table 1,[1] containing $T = 100$ samples each one. The experiments are presented in the next sections. In all of them, the SVM classifier was adopted. The parameters considered in Infomax and FastICA (e.g. functions for FastICA) were adjusted by numerical experiments.

3.1. The influence of the separation step in the classification

This first experiment aims at verifying the importance of separation step in the classification. In this context, we consider a problem of separating two

---

[1]In this table, the following notation is adopted: $s_i(t)$ is the value of the control chart i sampled at t ($t = 1, 2, \ldots, T$ is the time point), μ is the mean value of the control chart (we consider $\mu = 0$ in this study), $r_i(t)$ is a random number drawn from a standard normal distribution at the time point t, σ and σ' are control chart standard deviations (we consider $\sigma = 1$ in this study), d is the systematic deviation, a is the amplitude, T is the period (we consider $T = 16$ in this study), g is the gradient and s is the shift magnitude.

Table 1: Automatic pattern generating.

Patterns	Generator equation	Parameters
Normal	$s_i(t) = \mu + r_i(t)\sigma$	$\sigma = 1$
Stratification	$s_i(t) = \mu + r_i(t)\sigma'$	$\sigma' = 0.3\sigma$
Systematic	$s_i(t) = \mu + r_i(t)\sigma + d(-1)^t$	$d = 2\sigma$
Cyclic	$s_i(t) = \mu + r_i(t)\sigma + a\ sin(2\pi/T)$	$a = 2\sigma$
Increasing/decreasing trend	$s_i(t) = \mu + r_i(t)\sigma \pm tg\sigma$	$g = 0.075\sigma$
Upward/downward shift	$s_i(t) = \mu + r_i(t)\sigma \pm sk$	If $t > T/2$, $k = 0$. Else, $k = 1$. $s = 2\sigma$

sources $\mathbf{s}$, that is, two single control charts. We parametrize the retrieved control charts, $\mathbf{y}(t)$, as follows

$$\mathbf{y}(t) = \mathbf{Gs}(t), \tag{14}$$

where $\mathbf{G} \in \mathbb{R}^{2\times 2}$ is thus a global matrix associating the original sources (original single control charts) with the retrieved ones. In order to quantify the influence of the separation method in the classification, we estimate the correct classification rate for different values of $\mathbf{G}$. When $\mathbf{G}$ is close to a diagonal matrix, the separation is considered perfect (no mixture). Otherwise, the signals $\mathbf{y}(t)$ are still mixed versions of $\mathbf{s}(t)$, which means that concurrent charts are observed.

In our analysis the matrix $\mathbf{G}$ is parametrized by an angle θ to simulate

different cases of mixture. This parametrization is given by

$$\mathbf{G} = \begin{bmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{bmatrix},$$

where θ takes the values in the interval $[-\pi/2, \pi/2]$. For each value of θ, the values of $\mathbf{y}(t)$ are obtained and, then, they are submitted to the classifier. In order to have a better estimation, we conducted 1000 simulations for each value of θ and calculate its average.

The results obtained considering a mixture of normal and systematic patterns are shown in Figure 5. For a mixture of normal and cyclic patterns, the results are presents in Figure 6. In both analyses, we considered BSS-data and processed data (shape and statistical features) as the input of the classifier.

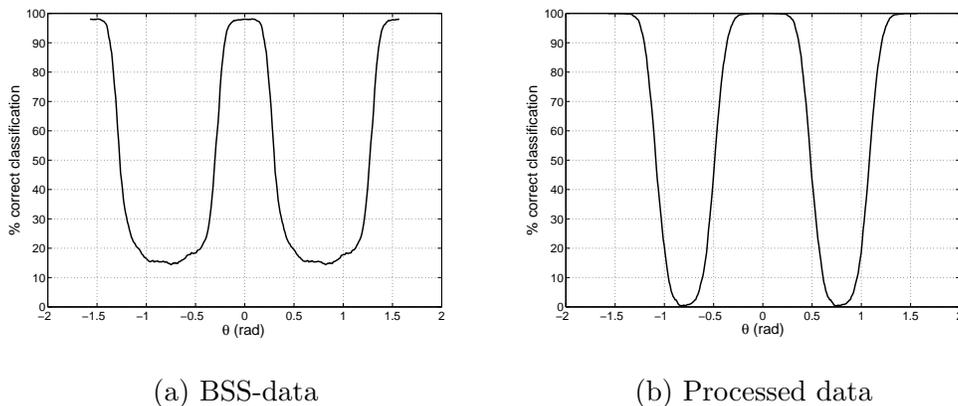(a) BSS-data   (b) Processed data

Figure 5: Classification's performance in the separation of normal/systematic mixture using different inputs, varying the global matrix.

As can be observed in Figures 5a and 5b, the classification rate of the normal/systematic mixture is higher in the cases where the separation is
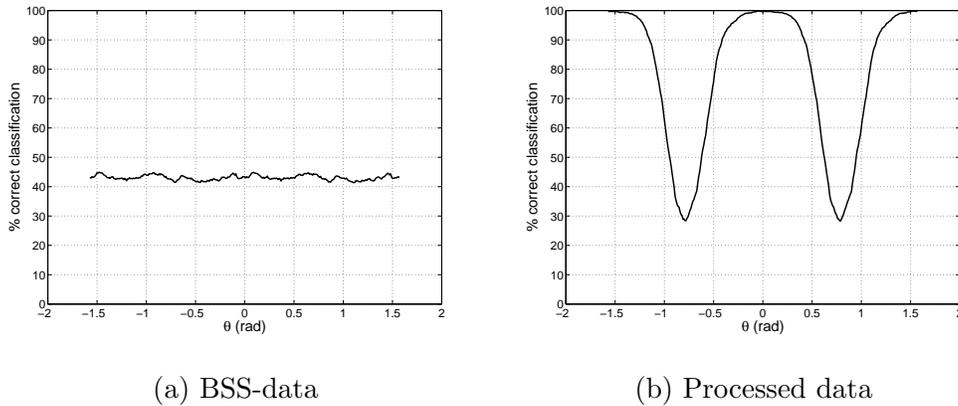
19

(a) BSS-data　　　　　　　　(b) Processed data

Figure 6: Classification's performance in the separation of normal/cyclic mixture using different inputs, varying the global matrix.

perfect (values of θ equals to $-\pi/2$, 0 and $\pi/2$) and it is low when θ assumes the values of $-\pi/4$ or $\pi/4$, cases where there is the maximum degree of mixture. If we consider the processed data as the input of the classifier (Figure 5b), the curve becomes narrow, highlighting that the use of shape and statistical features provides better classification rates. This is more evident when we analyze the normal/cyclic mixture. Since both patterns have similar spectrum, when BSS-data is used (Figure 6a), the classifier does not work well, with rates below 50%. Besides the use of shape and statistical features (Figure 6b) leads to 30% rates (θ around $\pm\pi/4$), the parameters can be discriminated by the classifier, also resulting in 100% rates.

The results highlight the relevance of conducting a BSS step before classification. Moreover, one can note that the feature extraction is also fundamental. Indeed, besides rendering classification more robust with respect to mixing, the feature extraction step allows one to achieve a good classification in the context of a normal/cyclic mixture — note that in this case the solely

20

application of BSS does not solve the problem.

3.2. Comparison between BSS methods in the separation step

Let us now compare the performance of different BSS methods in the separation step of control charts. In this experiment, at first, we study the separation of two control charts: a normal pattern and a given abnormal pattern. The mixture was generated by a linear process, as follows

$$x_i(t) = \mathbf{A}s_i(t) + r_i(t), \quad t = 1, \ldots, T, \tag{15}$$

where $s_i(t), i = 1, \ldots, n$ are the generated charts (original charts), $\mathbf{A}$ is the linear mixing process, $x_i, i = 1, \ldots, n$ are the mixture charts and r_i represents additive white Gaussian noise (AWGN). It is important to emphasize here that the range of Signal-to-Noise Ratio (SNR)^2 considered was $SNR = (0, 50]$ dB and the elements of $\mathbf{A}$ were

$$\mathbf{A} = \begin{bmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{bmatrix}.$$

The separation performance were computed base on the average value of the resulting signal-to-interference ratios (SIR):

$$SIR_i = 10 \log \left(\frac{E\{s_i(t)^2\}}{E\{(s_i(t) - y_i(t))^2\}} \right), \tag{16}$$

where $s_i(t)$ and $y_i(t)$ denote, respectively, the original source (single contro chart) and the separated signal (estimated control chart). For each method and each value of SNR, 1250 simulations was realized and the SIR value, averaged over the 1000 runs, was calculated.

---

^{2}The SNR is given by $SNR = 10 \log \sigma^2_{signal}/\sigma^2_{noise}$, where σ^2_{signal} is the signal power and σ^2_{noise} is the noise power.
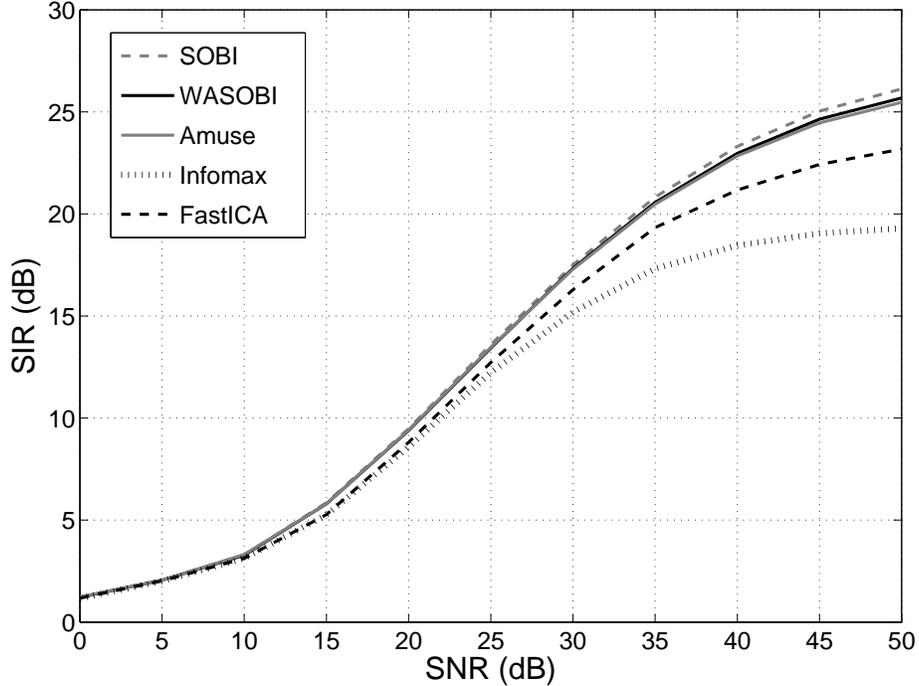
Figure 7: Error in the separation matrix when the mixtures are corrupted by AWG noise.

The results (Figure 7) show that the SOS methods (SOBI, WASOBI and AMUSE) perform better than HOS methods, especially for values of $SNR > 10$ dB.

We also compared the separation methods with respect to the computing time. The results can be seen in Table 2.[3] Again, SOS methods achieved better performance, especially the SOBI and AMUSE, due to the lower computational effort than the other methods.

---

[3]Computing device: Intel Core i7, 2.20 GHz, 8.00 GB RAM, software MATLAB 2009.

Table 2: Separation time.

Time (seconds)				
SOBI	WASOBI	AMUSE	Infomax	FastICA
0.0006	0.0033	0.0004	0.0924	0.3175

3.3. Comparison between BSS methods in the classification

The third experiment consists in comparing different BSS methods in the context of pattern classification, considering BSS-data and processed data as the inputs of the classifier. We compare the performances in both binary and multiclass classification.

In binary classification, at first, a set of 100 control charts of normal and 100 control charts of abnormal patterns was generated and trained by the classifier. After that, another set of control charts containing the same patterns used in training was generated and mixed by the linear process

$$x_i(t) = \mathbf{A} s_i(t), \tag{17}$$

as described in Section 3.3, without Gaussian noise. Then, for each mixture, a BSS method was applied in order to recover the original charts and, then, they were submitted to the SVM (BSS-data and processed data). The results obtained are shown in Tables 3 and 4.

Considering the results of Table 3, classification using SOS methods (SOBI, AMUSE and WASOBI) provides better results compared with HOS methods (Infomax and FastICA) when there is a trend pattern involved. This is due to the temporal structure observed in this pattern. The worst classification for all the BSS methods arose in the mixture of normal/cyclic,

23

Table 3: Binary pattern classification, using BSS-data as the input.

Mixed patterns	% of correct classification				
	SOBI	WASOBI	AMUSE	Infomax	FastICA
Normal/Systematic	80.4%	72.3%	63.9%	75.1%	58.7%
Normal/Cyclic	45.9%	45.4%	43.6%	46.0%	49.5%
Normal/Trend	100%	100%	100%	95.2%	94.3%
Normal/Shift	50.1%	51.3%	51.8%	53.5%	51.0%
Total average	69.10%	67.25%	64.82%	67.45%	63.38%

Table 4: Binary pattern classification, using processed data as the input.

Mixed patterns	% of correct classification				
	SOBI	WASOBI	AMUSE	Infomax	FastICA
Normal/Systematic	100%	100%	100%	100%	99.8%
Normal/Cyclic	100%	100%	100%	95.9%	95.1%
Normal/Trend	100%	99.9%	100%	97.8%	96.7%
Normal/Shift	99.6%	98.9%	99.1%	87.9%	90.2%
Total average	99.90%	99.70%	99.78%	95.40%	95.45%

once these patterns have almost the same spectrum and, then, it is not possible for the classifier to discriminate them. However, when using shape and statistical features as the input of the SVM (Table 4), the classifier can discriminate efficiently the normal/cyclic mixture, since the MAC value (see Equation (13)) for the cyclic pattern is higher than the normal one. As a consequence, the classification rates of this mixture is increased by using features instead of BSS-data.

In multiclass classification, i.e. when we aimed to distinguish 1 class among 5 possible ones, the training set was composed of 10 structures, one for

each possible mixture pair (considering all the combinations between normal, cyclic, systematic, trend and shift patterns, i.e. $C_{5,2}$ structures[4]). Similar to the binary case, the training structure case composed of 100 control charts of each pattern. Once we had the training structures, the mixed patterns were generated and submitted to the classifier. The results using BSS-data and processed data are presented in Table 5 and 6.

Table 5: Multiclass pattern classification, using BSS-data as the input.

Mixed patterns	% of correct classification				
	SOBI	WASOBI	AMUSE	Infomax	FastICA
Normal/Systematic	12.8%	10.0%	10.8%	25.9%	28.8%
Normal/Cyclic	48.0%	41.8%	50.3%	37.8%	46.1%
Normal/Trend	50.6%	50.4%	47.9%	47.6%	38.9%
Normal/Shift	52.6%	56.2%	48.0%	42.4%	39.8%
Total average	41.00%	39.60%	39.25%	38.42%	38.40%

These results show that the BSS-data do not lead to a satisfying classification rate in the multiclass classification problem. However, when the shape and statistical features are used, the efficiency of the SVM classification is highly improved, resulting in higher classification rates. Comparing the BSS methods, it can be noted that SOS methods provided better results with respect to HOS methods, always yielding a very low computational cost.

---

[4] $C_{5,2} = \frac{5!}{2!(5-2)!} = 10$ represents the combination of 5 different charts taken 2 at a time without repetition.

Table 6: Multiclass pattern classification, using processed data as the input.

Mixed patterns	% of correct classification				
	SOBI	WASOBI	AMUSE	Infomax	FastICA
Normal/Systematic	99.9%	99.7%	99.7%	99.4%	90.5%
Normal/Cyclic	99.5%	88.6%	85.6%	80.6%	59.0%
Normal/Trend	94.1%	94.0%	72.7%	71.5%	55.3%
Normal/Shift	97.4%	90.2%	97.3%	74.8%	84.7%
Total average	97.72%	93.12%	88.82%	81.58%	72.38%

3.4. Robustness in different non-randomness levels

Considering that SOBI method combined with shape and statistical features provided the better results in concurrent control chart pattern recognition, in this experiment, we verify the efficiency of this approach in situations of different non-randomness levels. For this, the parameters considered in pattern generation (Table 1) are modified in the testing dataset in order to consider different degrees of non-randomness levels in the charts. In respect of the training dataset, the parameters for the different situations were always the same as described in Table 1. Binary and multiclass pattern classification were applied as in the last experiment and the results are shown in Tables 7 and 8, respectively.

The results in binary classification confirms the robustness of the approach even in a smooth presence of a non-randomness. The worst classification rate (79.9%) was obtained considering normal/shift mixture, which is a high value for such degree of non-randomness.

In multiclass classification, a low level of non-randomness confuses the classifier, leading to lower correct recognition rates. However, when this

Table 7: Binary pattern classification for different non-randomness levels, using processed data as the input.

Mixed patterns	Parameter values	% of correct classification
Normal/Systematic	$d = 1\sigma$	99.3%
	$d = 1.5\sigma$	100%
	$d = 2\sigma$	100%
	$d = 2.5\sigma$	100%
	$d = 3\sigma$	100%
Normal/Cyclic	$a = 1\sigma$	86.5%
	$a = 1.5\sigma$	99.1%
	$a = 2\sigma$	100%
	$a = 2.5\sigma$	100%
	$a = 3\sigma$	100%
Normal/Trend	$g = 0.025\sigma$	91.0%
	$g = 0.050\sigma$	99.1%
	$g = 0.075\sigma$	100%
	$g = 0.1\sigma$	100%
	$g = 0.125\sigma$	100%
Normal/Shift	$s = 1\sigma$	79.9%
	$s = 1.5\sigma$	95.6%
	$s = 2\sigma$	99.6%
	$s = 2.5\sigma$	99.9%
	$s = 3\sigma$	100%

level increases, the correct classification rates also increase, specially for normal/systematic and normal/trend mixtures. We also can note, for normal/cyclic and normal/shift mixtures, that a deviation from the parameter used in training step leads to incorrect classification.

Table 8: Multiclass pattern classification for different non-randomness levels, using processed data as the input.

Mixed patterns	Parameter values	% of correct classification
Normal/Systematic	$d = 1\sigma$	49.4%
	$d = 1.5\sigma$	98.9%
	$d = 2\sigma$	99.9%
	$d = 2.5\sigma$	99.5%
	$d = 3\sigma$	99.9%
Normal/Cyclic	$a = 1\sigma$	7.5%
	$a = 1.5\sigma$	63.9%
	$a = 2\sigma$	99.5%
	$a = 2.5\sigma$	80.8%
	$a = 3\sigma$	59.9%
Normal/Trend	$g = 0.025\sigma$	1.0%
	$g = 0.050\sigma$	38.9%
	$g = 0.075\sigma$	94.1%
	$g = 0.1\sigma$	99.7%
	$g = 0.125\sigma$	99.9%
Normal/Shift	$s = 1\sigma$	64.6%
	$s = 1.5\sigma$	95.4%
	$s = 2\sigma$	97.4%
	$s = 2.5\sigma$	66.0%
	$s = 3\sigma$	80.9%

3.5. Robustness to imbalanced dataset

Also considering SOBI method combined with shape and statistical features, in this experiment, we verify its performance when the dataset used in training step is imbalanced. Differently from the Section 3.3 where we used an equal proportion of normal and unnatural pattern (100/100), here

this ratio is increased, considering the more normal dataset than unnatural, which is expected in a practical situation. The results for binary pattern classification is shown in Table 9.

Table 9: Binary pattern classification for imbalanced data, using shape and statistical features as the input.

Ratio dataset (normal/unnatural patterns)	% of correct classification			
	Normal/ Systematic	Normal/ Cyclic	Normal/ Trend	Normal/ Shift
100/100	100%	100%	100%	99.4%
120/80	100%	100%	100%	99.3%
140/60	100%	99.9%	100%	98.8%
160/40	100%	100%	100%	99.3%
180/20	100%	99.9%	100%	97.9%
190/10	100%	99.9%	100%	95.1%
192/8	100%	100%	100%	92.5%
194/6	100%	99.5%	100%	96.3%
196/4	100%	100%	100%	98.7%
198/2	100%	99.6%	100%	87.6%

The results show the robustness of the approach even with a high ratio normal/unnatural dataset. It is due to the classifier to consider only two patterns in the traning step and, then, there is no confusion with other possible patterns. The worst classification rate (87.6%) was obtained considering normal/shift mixture, which is a high value for such ratio normal/unnatural dataset.

4. Conclusions

Given the importance of detecting failures in the production process, it is necessary to use tools that can efficiently recognize abnormalities involved in. This work deals with concurrent control charts pattern classification, which the detection is hindered due to the mixture of the individual characteristics. In this context, a better comprehension and comparison between existing methods is crucial to verify which one provides better results in separation and classification steps. In that respect, a first experiment performed in this work showed the high influence that the separation and feature extraction step has in the classification, justifying our purpose to compare different BSS methods and a selection of shape and statistical feature in order to verify which one provides higher classification rates.

In a second experiment, we showed that SOS-based BSS methods perform better than HOS-based methods (ICA) and with a very low computational cost (with a gain of about 2 orders of magnitude). It is worth noting that most of current works that deal with BSS in control charts consider HOS-based methods. Furthermore, the results have shown the higher accuracy when the input of the classifier is based on a set of shape and statistical features, as these parameters offer a better discrimination of patterns involved in control charts.

Based on these results, we can conclude that the approach composed by a SOS method (especially SOBI and WASOBI) and shape and statistical features provides better results in concurrent control chart pattern recognition. SOS-based methods insure an efficient separation of the mixed control charts and the shape/statistical features extracted by the estimated sources

provides parameters that are useful in the classification step even when the solely application of separation is not enough to assure a good classification. Fourth and fifth experiments also shown the robustness of this approach even with different non-randomness levels and imbalanced data, specially for normal/systematic mixture.

In future researches, we aim to verify the performance of the SOS methods combined with shape and statistical in real industrial environments. Furthermore, we aim to apply the proposed approach in online classification, where non-randomness arises progressively.

Acknowledgements

References

[1] D. C. Montgomery, G. C. Runger, Applied statistics and probability for engineers, 3rd Edition, John Wiley & Sons, 2003.

[2] D. C. Montgomery, Introduction to statistical quality control, 5th Edition, John Wiley & Sons, Hoboken, NJ, USA, 2005.

[3] W. E. Company, Statistical quality control handbook, Western Electric, Indiana, 1958.

[4] L. S. Nelson, The shewhart control chart - tests for special causes, Journal of Quality Technology 16 (4) (1984) 237–239.

[5] H.-P. Cheng, C.-S. Cheng, Control chart pattern recognition using wavelet analysis and neural networks, Journal of Quality 16 (5) (2009) 311–321.

[6] V. Ranaee, A. Ebrahimzadeh, Control chart pattern recognition using a novel hybrid intelligent method, Applied Soft Computing 11 (2011) 2676–2686.

[7] D. T. Pham, M. A. Wani, Feature-based control chart pattern recognition, International Journal of Production Research 35 (7) (1997) 1875–1890.

[8] V. Ranaee, A. Ebrahimzadeh, R. Ghaderi, Application of the pso–svm model for recognition of control chart patterns, ISA Transactions 49 (2010) 577–586.

[9] S. K. Gauri, Control chart pattern recognition using feature-based learning vector quantization, International Journal of Advanced Manufacturing Technology 48 (2010) 1061–1073.

[10] S. K. Gauri, S. Chakraborty, Recognition of control chart patterns using improved selection of features, Computers & Industrial Engineering 56 (2009) 1577–1588.

[11] M. Bag, S. K. Gauri, S. Chakraborty, Recognition of control chart patterns using discriminant analysis of shape features, in: International Conference on Industrial Engineering and Operations Management, 2010.

[12] M. Zhang, W. Cheng, Recognition of mixture control chart pattern using multiclass support vector machine and genetic algorithm based on statistical and shape features, Mathematical Problems in Engineering.

[13] C. Wu, F. Liu, B. Zhu, Control chart pattern recognition using an integrated model based on binary-tree support vector machine, International Journal of Production Research 53 (7) (2015) 2026–2040.

[14] P. Xanthopoulos, T. Razzaghi, A weighted support vector machine method for control chart pattern recognition, Computers & Industrial Engineering 70 (2014) 134–149.

[15] R.-S. Guh, J. D. T. Tannock, Recognition of control chart concurrent patterns using a neural network approach, International Journal of Production Research 37 (8) (1999) 1743–1765.

[16] W.-A. Yang, W. Zhou, W. Liao, Y. Guo, Identification and quantification of concurrent control chart patterns using extreme-point symmetric mode decomposition and extreme learning machines, Neurocomputing 147 (2015) 260–270.

[17] Z. Chen, S. Lu, S. Lam, A hybrid system for spc concurrent pattern recognition, Advanced Engineering Informatics 21 (2007) 303–310.

[18] J.-H. Yang, M.-S. Yang, A control chart pattern recognition system using a statistical correlation coefficient method, Computers & Industrial Engineering 48 (2005) 205221.

[19] S. Du, D. Huang, J. Lv, Recognition of concurrent control chart patterns

using wavelet transform decomposition and multiclass support vector machines, Computers & Industrial Engineering 66 (2013) 683–695.

[20] N. Gu, Z. Cao, L. Xie, D. Creighton, M. Tan, S. Nahavandi, Identification of concurrent control chart patterns with singular spectrum analysis and learning vector quantization, Journal of Intelligent Manufacturing 24 (2013) 1241–1252.

[21] L. Xie, N. Gu, D. Li, Z. Cao, M. Tan, S. Nahavandi, Concurrent control chart patterns recognition with singular spectrum analysis and support vector machine, Computers & Industrial Engineering 64 (2013) 280–289.

[22] A. Hyvärinen, J. Karhunen, E. Oja, Independent component analysis, John Wiley & Sons, 2001.

[23] P. Comon, C. Jutten, Handbook of blind source separation: independent component analysis and applications, Academic Press, 2010.

[24] C.-H. Wang, T.-P. Dong, W. Kuo, A hybrid approach for identification of concurrent control chart patterns, Journal of Intelligent Manufacturing 20 (2009) 409–419.

[25] C.-J. Lu, Y. E. Shao, P.-H. Li, Mixture control chart patterns recognition using independent component analysis and support vector machine, Neurocomputing 74 (2011) 1908–1914.

[26] C. Jutten., J. Hérault, Blind separation of sources, part i - an adaptative algorithm based on neuromimetic architecture, Signal Processing 24 (1991) 1–10.

[27] G. Darmois, Analyse générale des liaisons stochastiques, Revue de l'Institut International de Statistique 21 (1/2) (1953) 2–8.

[28] P. Comon, Independent component analysis, a new concept?, Signal Processing 36 (1994) 287–314.

[29] J. F. Cardoso, Infomax and maximum likelihood for blind source separation, IEEE Signal Processing Letters 4 (4) (1997) 112–114.

[30] J. F. Cardoso, Blind signal separation: statistical principles, in: Proceedings of the IEEE, Vol. 86, 1998, pp. 2009–2025.

[31] A. J. Bell, T. J. Sejnowski, An information-maximization approach to blind separation and blind deconvolution, Neura 7 (1995) 1129–1159.

[32] K. T. Herring, A. V. Mueller, D. H. Staelin, Blind separation of noisy multivariate data using second-order statistics: remote-sensing applications, IEEE Transactions on Geoscience and Remote Sensing 47 (10) (2009) 3406–3415.

[33] F. Ghaderi, H. R. Mohseni, S. Sanei, A fast second order blind identification method for separation of periodic sources, in: 18th European Signal Processing Conference, 2010, pp. 1572–1576.

[34] L. Tong, V. C. Soon, Y. F. Huang, R. Liu, Amuse: A new blind identification algorithm, Circuits and Systems, 1990., IEEE International Symposium (1990) 1784–1787.

[35] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, E. Moulines, A blind

source separation technique using second-order statistics, IEEE Transactions on Signal Processing 45 (2) (1997) 434–444.

[36] A. Yeredor, Blind separation of gaussian sources via second-order statistics with asymptotically optimal weighting, IEEE Signal Processing Letters 7 (7) (2000) 197–200.

[37] C.-W. Hsu, C.-J. Lin, A comparison of methods for multiclass support vector machines, IEEE Transactions on Neural Networks 13 (2) (2002) 415–425.