A ROBUST CLASSIFICATION METHOD USING COMBINED CLASSIFIERS IN A NONSTATIONARY ENVIRONMENT

Yuan Dong, Pierre Beauseroy and André Smolarz

Institut Charles Delaunay - LM2S - UMR STMR 6279 CNRS
Université de Technologie de Troyes
12 Rue Marie Curie, BP2060, 10010 Troyes Cedex, France

ABSTRACT

In this paper, we present a robust data classification method based on an ensemble of feature subspaces. The objective is to improve or preserve the performances of a decisional system in the case of perturbations due to noise or sensor degradation. The proposed method is to combine a set of classifiers each of which is established in the corresponding feature subspace resulting from projections of the initial full-dimensional space, expecting that most of them are not impaired. The counterpart of the expected robustness is a performance decrease for non-impaired data. In this context, three classification methods are tested, One-class SVM, Kernel PCA and Kernel ECA, to study the robustness of the final decision. The results obtained in textured image segmentation demonstrate that our approach is efficient in a nonstationary environment.

***Index Terms*—** One-class classification, ensemble method, decision, One-class SVM, Kernel PCA, Kernel ECA, textured image segmentation

1. INTRODUCTION

The diagnosis or monitoring of a complex system can be considered as a classification problem. When no analytical model of the system is known, the decision rule can be learned based on a training set of samples coming from the system. Each sample is represented by a feature vector obeying a joint probability distribution that depends on the state of the system. Then the established decision rule is used to classify new data. In general, the decision rule performs well if the new data obeys the joint distribution of the training samples. In practice, one part of the system measurements may be perturbed due to the presence of noise or to the failure of some sensors. As a consequence, the feature vector no longer obeys the trained distribution. In such situation the established decision rule can not guarantee to classify data correctly and its performance may significantly decreases.

In order to preserve the performance of the decision rule under serious perturbations, the proposed method is to make the final decision by combining a set of classifiers each of which is established in a corresponding feature subspace. These feature subspaces are defined using a restricted part of the system measurements, expecting that most of them are unperturbed. There are many methods to generate feature subspaces in the literature, such as the random subspace method [1], the random feature-subset selection in a nonstationary environment [2] and so on. And the ensemble method [3] is an appropriate method to implement this idea.

In this study the monitoring problem is considered as a one-class decision problem. For most of one-class classification methods, the main idea is to train a model so that its response to a sample drawn from the learn class obeys some specific condition (*i.e.* the response is positive). In recent years, the theory of SVM (Support Vector Machine) is the most widely used, of which One-class SVM proposed by Schölkopf et al. [4] is very suitable to tackle the one-class classification problem. The objective of One-class SVM is to find an optimum hyperplane to separate the data from the origin in feature space. Another kind of method is Kernel PCA (principal component analysis) [5] using the kernel trick to handle the nonlinear data distribution. Its strategy is to extract the principal axes of the data distribution in feature space and to use the squared distance to these principal axes as a novelty measure. Hoffmann [6] has demonstrated that this method has lower classification errors and tighter decision boundary than One-class SVM. Similar to Kernel PCA, Jenssen proposed a new method named Kernel ECA (entropy component analysis) [7] based on the Renyi entropy of the input data and their results are also very encouraging. In this paper, we will test the three methods and compare their performances.

This paper is organized as follows : Section 2 briefly reviews the principals of classification methods (One-class SVM, Kernel PCA and Kernel ECA) and the definition of novelty measure. Section 3 presents the proposed method based on feature subspaces and the testing procedure for the study of robustness. Section 4 reports the experiments and results, and the study is concluded in Section 5.

This work is partially supported by China Scholarship Council of Chinese Government.

2. PRINCIPLES OF CLASSIFICATION METHODS

2.1. One-class SVM (OSVM)

Suppose that $\mathcal{X}^m \subseteq \mathbb{R}^m$ is an initial representation space of dimension m and $\mathcal{A}_n = \{\boldsymbol{x}_i \in \mathcal{X}^m | i = 1, \ldots, n\}$ is a set of n training samples. The principle of One-class SVM proposed by Schölkopf et al.[4] is to define a function f that takes the value $+1$ in a subspace $S \subset \mathcal{X}^m$ capturing most of the samples drawn from an unknown probability distribution and -1 elsewhere. Its strategy is first to map the training samples $\mathcal{A}_n$ into the feature space by a nonlinear transformation ϕ for which the inner product of its images can be computed by evaluating some simple kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$, such as the gaussian kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$ that is widely used. Unless otherwise noted, a gaussian kernel with width σ has been used with all studied methods. The function is defined so that one part of the samples (with a proportion of $1-\nu$) are separated from the origin, in the feature space, by a hyperplane with maximum margin.

In order to determine the maximum margin hyperplane, we need to deduce its normal vector $\mathbf{w}$ and a threshold ρ by solving the following quadratic program :

$$\begin{cases} \min_{\mathbf{w},\xi,\rho} : & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{\nu n}\sum_{i=1}^n \xi_i - \rho \\ \text{subject to :} & \langle \mathbf{w}, \phi(\boldsymbol{x}_i) \rangle \geq \rho - \xi_i, \quad \xi_i \geq 0 \end{cases} \tag{1}$$

The dual problem can be expressed as follows :

$$\begin{cases} \min_{\alpha} : & \frac{1}{2}\sum_{i,j=1}^n \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ \text{subject to :} & 0 \leq \alpha_i \leq \frac{1}{\nu n}, \quad \sum_{i=1}^n \alpha_i = 1 \end{cases} \tag{2}$$

and the decision function is given by :

$$\begin{aligned} f(\boldsymbol{x}) &= sign\Big(\sum_{i=1}^n \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}) - \rho\Big) \\ &= sign\big(\langle \mathbf{w}, \phi(\boldsymbol{x}) \rangle - \rho\big) \end{aligned} \tag{3}$$

2.2. Kernel PCA (KPCA)

KPCA [5] can be seen as a nonlinear extension of traditional PCA [8]. The first stage is also to map the training samples $\mathcal{A}_n$ into the feature space by a nonlinear transformation ϕ, and then the traditional PCA is performed. A simple kernel is used again to solve the problem of inner product, hence an eigenvector $\mathbf{v}$ of the covariance matrix in this feature space is expressed by :

$$\mathbf{v} = \sum_{i=1}^n \tilde{v}_i \tilde{\Phi}(\boldsymbol{x}_i) \tag{4}$$

with

$$\tilde{\Phi}(\boldsymbol{x}_i) = \phi(\boldsymbol{x}_i) - \frac{1}{n}\sum_{r=1}^n \phi(\boldsymbol{x}_r) \tag{5}$$

There $\tilde{v}_i$ are the components of a vector $\tilde{\mathbf{v}}$ that is an eigenvector of the $(n \times n)$ matrix $\tilde{\mathbf{K}}$ defining by $\tilde{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \tilde{\Phi}(\boldsymbol{x}_i), \tilde{\Phi}(\boldsymbol{x}_j) \rangle$. Requiring $\|\mathbf{v}\| = 1$, we thus have $\mathbf{v} = \lambda^{-\frac{1}{2}} \sum_{i=1}^n \tilde{v}_i \tilde{\Phi}(\boldsymbol{x}_i)$, where λ is the eigenvalue of $\tilde{\mathbf{K}}$ corresponding to $\tilde{\mathbf{v}}$.

2.3. Kernel ECA (KECA)

KECA, developed by Jenssen [7], is a new data transformation method based on the Renyi quadratic entropy [9] relative to KPCA. In KECA, the data transformation is performed by projecting onto these KPCA axes not corresponding to the top eigenvalues or eigenvectors of the kernel matrix, but contributing more to the entropy estimate.

The Renyi entropy is given by $H(p) = -\log \int p^2(\boldsymbol{x}) d\boldsymbol{x}$. Since the logarithm is a monotonic function, we can concentrate on the quantity $V(p) = \int p^2(\boldsymbol{x}) d\boldsymbol{x} = E[p(\boldsymbol{x})]$, where $p(\boldsymbol{x})$ is the probability density of the training set $\mathcal{A}_n$ and it can be evaluated by $\hat{p}(\boldsymbol{x}) = \frac{1}{n}\sum_{\boldsymbol{x}_i \in \mathcal{A}_n} K_\sigma(\boldsymbol{x}, \boldsymbol{x}_i)$ (Parzen kernel estimator if K_σ is adequately normalized). Hence $\hat{V}(p)$ can be estimated by,

$$\begin{aligned} \hat{V}(p) &= \frac{1}{n}\sum_{\boldsymbol{x}_i \in A_n} \hat{p}(\boldsymbol{x}_i) = \frac{1}{n^2}\mathbf{1}^T\mathbf{K}\mathbf{1} \\ &= \frac{1}{n^2}\mathbf{1}^T\mathbf{E}\Lambda\mathbf{E}^T\mathbf{1} = \sum_{i=1}^n (\sqrt{\hat{\lambda}_i}\mathbf{e}_i^T\mathbf{1})^2 \end{aligned} \tag{6}$$

where $\mathbf{1}$ is a all-ones vector $(n \times 1)$, and $\mathbf{K}$ is the kernel matrix $(n \times n)$ with general term $K_\sigma(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$, which is eigendecomposed as $\mathbf{K} = \mathbf{E}\Lambda\mathbf{E}^T$.

2.4. Reconstruction error

The methods KPCA and KECA could be applied in the one-class data classification task. Herein the reconstruction error ([10] and [6]) is proposed as a novelty measure. The decision boundaries are determined by the equipotential lines or surfaces of the reconstruction error computed in the feature space. The reconstruction error is defined by :

$$\begin{cases} \varepsilon^{KPCA} = \langle \tilde{\Phi} \cdot \tilde{\Phi} \rangle - \langle \tilde{U}\tilde{\Phi} \cdot \tilde{U}\tilde{\Phi} \rangle \\ \varepsilon^{KECA} = \langle \phi \cdot \phi \rangle - \langle U\phi \cdot U\phi \rangle \end{cases} \tag{7}$$

where $\tilde{\mathbf{U}}$ contains q eigenvectors corresponding to the q largest eigenvalues of $\tilde{\mathbf{K}}$, and $\mathbf{U}$ contains the q eigenvectors of $\mathbf{K}$ which contribute most to Renyi entropy. The projection of a new observation $\boldsymbol{x}$ onto the j^{th} eigenvector $\tilde{\mathbf{u}}_j$ (resp. $\mathbf{u}_j$) is then measured by,

$$\begin{cases} \epsilon_j^{KPCA}(\boldsymbol{x}) = \langle \tilde{\Phi}(\boldsymbol{x}), \tilde{\mathbf{u}}_j \rangle \\ \epsilon_j^{KECA}(\boldsymbol{x}) = \langle \phi(\boldsymbol{x}), \mathbf{u}_j \rangle, \qquad j = 1, \ldots, q \end{cases} \tag{8}$$

Combining equations 7 and 8, we can get the expressions of reconstruction errors for a new observation $\boldsymbol{x}$,

$$\begin{cases} \varepsilon^{KPCA}(\boldsymbol{x}) & = \tilde{K}(\boldsymbol{x}, \boldsymbol{x}) - \sum_{j=1}^q \left(\epsilon_j^{KPCA}(\boldsymbol{x})\right)^2 \\ \varepsilon^{KECA}(\boldsymbol{x}) & = K(\boldsymbol{x}, \boldsymbol{x}) - \sum_{j=1}^q \left(\epsilon_j^{KECA}(\boldsymbol{x})\right)^2 \end{cases} \tag{9}$$

The classification of $\boldsymbol{x}$ is then done by comparing $\varepsilon^{KPCA}(\boldsymbol{x})$ (resp. $\varepsilon^{KECA}(\boldsymbol{x})$) with a threshold.

3. PROPOSED METHOD

As mentioned in section 1, the proposed method is based on making decisions in some feature subspaces in which most of features are unperturbed so that corresponding classifier performances are preserved in the case of perturbed or nonstationary environment. First, L feature subspaces are generated by randomly selecting d features from m features in the full-dimension space $\{\mathcal{X}_\ell^d \subset \mathcal{X}^m | \ell = 1, \ldots, L\}$. The value of d can be randomly chosen for each subspace or determined by experience. Then we build a classifier h_ℓ corresponding to each feature subspace $\mathcal{X}_\ell^d$ and finally the final decision is obtained by combining these classifiers. Therefore, the final decision rule $D(\boldsymbol{x})$ for an observation $\boldsymbol{x}$ can be simply written as,

$$D(\boldsymbol{x}) = sign\big(\sum_{\ell=1}^{L} h_\ell(\boldsymbol{x}^{(\ell)}) - \theta\big) \tag{10}$$

where $h_\ell(\boldsymbol{x}^{(\ell)})$ are established respectively according to OSVM, KPCA and KECA,

$$\begin{cases} h_\ell^{OSVM}(\boldsymbol{x}^{(\ell)}) = \langle \mathbf{w}^{(\ell)}, \phi(\boldsymbol{x}^{(\ell)}) \rangle \\ h_\ell^{KPCA}(\boldsymbol{x}^{(\ell)}) = \varepsilon^{KPCA}(\boldsymbol{x}^{(\ell)}) \\ h_\ell^{KECA}(\boldsymbol{x}^{(\ell)}) = \varepsilon^{KECA}(\boldsymbol{x}^{(\ell)}) \end{cases} \tag{11}$$

θ is a threshold. The robustness of final decision $D(\boldsymbol{x})$ will be tested through the following procedure.

We note ω_0 as the normal class and ω_1 as the novel class. We construct a training set $\mathcal{A}_n$ (consisting of n observations from ω_0), two sets $\mathcal{T}_{n_0}$ (consisting of n_0 observations from ω_0) and $\mathcal{T}_{n_1}$ (consisting of n_1 observations from ω_1). And we note that,

$$\begin{cases} D(\boldsymbol{x}) = 1 & \text{label } \omega_0 \text{ is attributed to } \boldsymbol{x} \\ D(\boldsymbol{x}) = -1 & \text{label } \omega_1 \text{ is given to } \boldsymbol{x} \end{cases}$$

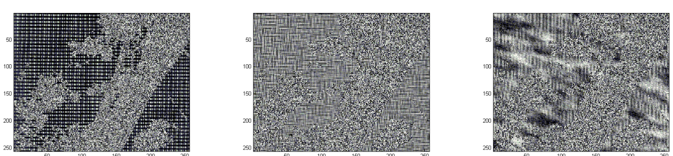
Since the testing procedures of these three methods are similar, here we consider only OSVM as example.

- For a given couple of parameter (ν, σ)
  1. To train the classifiers h_ℓ^{OSVM} :
     $\alpha_i^{(\ell)}$ and thus $\mathbf{w}^{(\ell)}$ were learned in $\mathcal{A}_n^{(\ell)}$ (associated to the feature subspace $\mathcal{X}_\ell^d$) for $\ell = 1, \ldots, L$
  2. To determine the threshold θ :
     $\forall\, \boldsymbol{x}_{k_0} \in \mathcal{T}_{n_0}$, the threshold $\theta = \rho_\alpha(\nu, \sigma)$ is found such that $D(\boldsymbol{x}_{k_0}) = sign\big(\sum_{\ell=1}^{L} h_\ell^{OSVM}(\boldsymbol{x}_{k_0}^{(\ell)}) - \rho_\alpha(\nu, \sigma)\big)$ satisfies a reject rate $\alpha = P[D(\boldsymbol{x}_{k_0}) = \omega_1 / \boldsymbol{x}_{k_0} \in \omega_0]$ fixed in advance.
  3. To estimate the error rate β :
     $\forall\, \boldsymbol{x}_{k_1} \in \mathcal{T}_{n_1}$, the rule $D(\boldsymbol{x}_{k_1})$ with $\theta = \rho_\alpha(\nu, \sigma)$ found above is applied to estimate the error rate $\beta_\alpha(\nu, \sigma) = P[D(\boldsymbol{x}_{k_1}) = \omega_0 / \boldsymbol{x}_{k_1} \in \omega_1]$
- To search the optimum parameter (ν^*, σ^*) such that $\beta_\alpha(\nu^*, \sigma^*) = \min_{\nu,\sigma}(\beta_\alpha(\nu, \sigma))$ under fixed α. The above-step were repeated to perform a grid search for different values of (ν, σ). Then $(\nu^*, \sigma^*, \theta^*)$ are used with the following test images.

For the methods KPCA and KECA based on the reconstruction error, the optimum parameter to be found is (q, σ). The threshold is the upper bound of the reconstruction error that is determined so that $\theta^* = \rho_\alpha^*(\nu^*, \sigma^*)$ under α fixed. The performances are finally studied on new data sets which are independent of $\mathcal{A}_n$, $\mathcal{T}_{n_0}$ and $\mathcal{T}_{n_1}$.

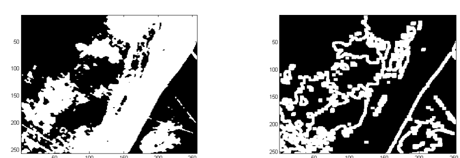
4. EXPERIMENTS

The textured image segmentation is an appropriate application for evaluating our proposed method because it effectively simulates the nonstationary environment at the boundaries. All the test images (Fig.1) are of size 256×256, and each of them is composed of the texture and the uniform noise. Fig.2-a illustrates the reference image of the ideal segmentation : black area is the texture representing the normal class ω_0 and white area is the noise ω_1. The two areas ω_0 and ω_1 are separated by complex boundaries. Fig.2-b illustrates the reference image of the different regions : central region (in blank) and boundary region (in white). The textures to be tested are extracted from the Brodatz album [11] (Fig.1-a) or generated by using Markov fields models [12] (Fig.1-b and 1-c).



a b c

Fig. 1: Test images



a b

Fig. 2: Reference images

In the case of textured image segmentation, the data to be classified are the pixels of the image. Each pixel can be described by a feature vector whose components are the gray levels of pixels located in a given neighborhood of this pixel. In the following experiments, a squared window 5×5 centered on the pixel considered is selected as the initial representation space $\mathcal{X}^m$ ($m = 5 \times 5 = 25$). We select randomly $L = 120$ feature subspaces which is composed of L_3 feature subspaces of dimension $d = 3$, L_5 feature subspaces of dimension $d = 5$ and L_7 feature subspaces of dimension $d = 7$ among $m = 25$ in $\mathcal{X}^m$. Three groups of (L_3, L_5, L_7) are tested respectively, they are (20, 40, 60), (40, 40, 40) and (60, 40, 20).

In order to illustrate the advantage of the proposed method based on the feature subspaces, we also applied the OSVM decision rule based on the initial full-dimension space $\mathcal{X}^m$ for comparison. The application and testing procedure are the same as above-presented in section 3. This method will be denoted OSVM_$\mathcal{X}^m$ while the method based on the subspaces will be denoted simply OSVM.

The four methods that need to be evaluated for their performances are OSVM_$\mathcal{X}^m$, OSVM, KPCA and KECA. For each test image, the pixels of image are distinguished in two types : one located in the central region (in black in Fig.2-a) and the other located in the boundary region (in white in Fig.2-a). These two types of pixels are defined by,

– Central region : a set of pixels whose all 24 neighbors belong to the same class as itself.
– Boundary region : a set of pixels whose one part of neighbors are from the other class as compared to itself.

The performance of each method is evaluated by estimating four probabilities of errors as following :

$$\begin{cases} \alpha_C = P[D(\boldsymbol{x}) = \omega_1 / \boldsymbol{x} \in \text{Central} \cap \omega_0] \\ \beta_C = P[D(\boldsymbol{x}) = \omega_0 / \boldsymbol{x} \in \text{Central} \cap \omega_1] \end{cases}$$

and
$$\begin{cases} \alpha_B = P[D(\boldsymbol{x}) = \omega_1 / \boldsymbol{x} \in \text{Boundary} \cap \omega_0] \\ \beta_B = P[D(\boldsymbol{x}) = \omega_0 / \boldsymbol{x} \in \text{Boundary} \cap \omega_1] \end{cases}$$

The Table 1 and Table 2 represent these four probabilities of errors for each test image and for each method. The parameters and thresholds of the rules were calculated by the sets $\mathcal{A}_n$, $\mathcal{T}_{n_0}$, $\mathcal{T}_{n_1}$ with the size of $n = 500$, $n_0 = 3000$, $n_1 = 10000$ under $\alpha = 0.01$ and $\alpha = 0.05$ fixed (Cf. section 3).

For both $\alpha = 0.01$ and $\alpha = 0.05$, according to the values of the error rate α_C and β_C in the central region, we can observe that the method OSVM_$\mathcal{X}^m$ is usually more efficient than those based on the subspaces. In contrast, in the boundary regions the methods based on the subspaces become more efficient with regards to the decision of the texture (normal) class (Cf. the values of α_B), in particular the methods KPCA and KECA. Besides, the performance of each method based on the subspaces is related to the proportion of (L_3,L_5,L_7), the results illustrate that in boundary regions, with the increase of proportion of low-dimensional feature subspaces (L_3), the error rates α_B decrease while the error rates β_B increase. The increase of proportion of high-dimensional feature subspaces (L_7) leads to contrary results. Figure 3 shows the best results in terms of α_B.

5. CONCLUSION

In this paper, we presented a robust data classification method combining a set of classifiers associated to the corresponding feature subspaces. The performance of the proposed method has been tested on the textured image segmentation, based on three classification methods One-class SVM, Kernel PCA and Kernel ECA respectively. The results indicated that the methods based on the subspaces preserve better the performance of decision compared with those based on the initial space, since the initial space no longer correctly represents the data in the boundary region due to the presence of perturbations. While the counterpart of the robustness has a relative performance decrease for unperturbed data (*i.e.* in the central region). In addition, the dimension of subspaces (the value of d) and the number of subspaces of each dimension can be chosen to adjust the tradeoff between the error rates β_B and α_B in boundary or perturbed regions.

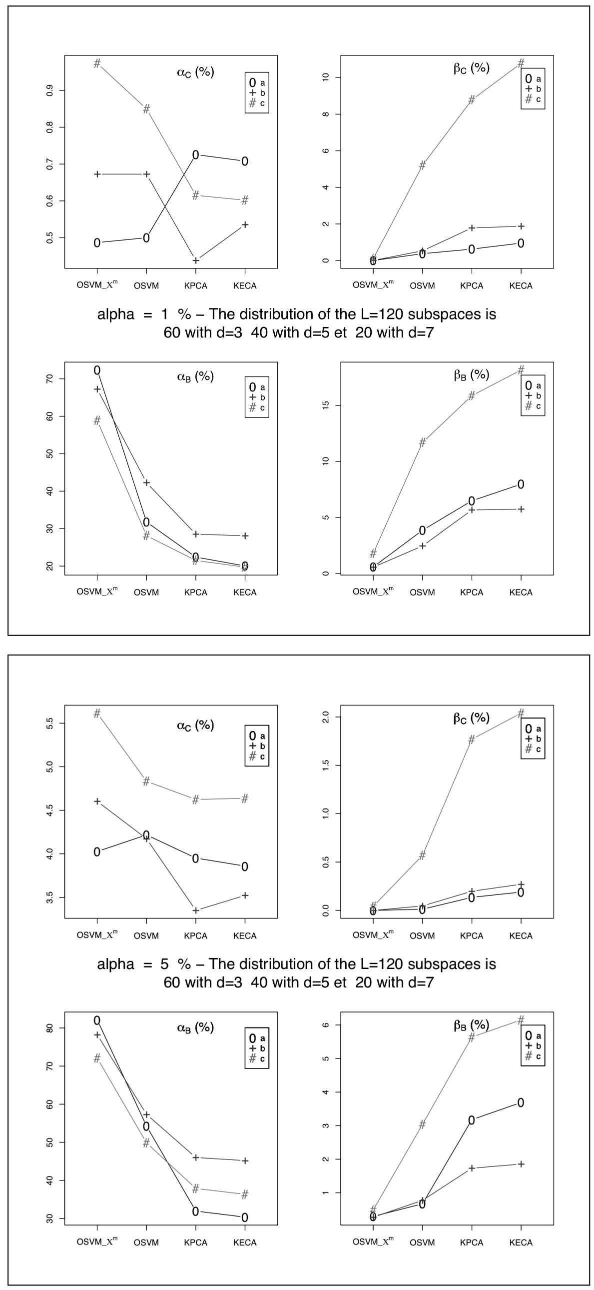


Fig. 3: Best results on images a, b and c of Fig.1 in terms of α_B for $\alpha = 1\%$ and 5%

Table 1: The error rate (%) with $\alpha = 0.01$

Image Fig.1-a	(L_3,L_5,L_7)	α_C	β_C	α_B	β_B
OSVM_$\mathcal{X}^m$		0.49	**0.00**	72.36	**0.58**
OSVM	(20,40,60)	**0.46**	**0.06**	37.62	**2.44**
	(40,40,40)	**0.46**	0.17	33.93	2.97
	(60,40,20)	0.50	0.38	**31.77**	3.86
KPCA	(20,40,60)	**0.69**	**0.13**	27.45	**4.36**
	(40,40,40)	0.73	0.29	25.43	5.13
	(60,40,20)	0.73	0.62	**22.41**	6.49
KECA	(20,40,60)	**0.70**	**0.17**	26.31	**4.74**
	(40,40,40)	**0.70**	0.39	23.92	5.83
	(60,40,20)	0.71	0.95	**20.03**	7.98
Image Fig.1-b	(L_3,L_5,L_7)	α_C	β_C	α_B	β_B
OSVM_$\mathcal{X}^m$		0.67	**0.00**	67.24	**0.56**
OSVM	(20,40,60)	0.65	**0.06**	49.17	**1.22**
	(40,40,40)	**0.53**	0.19	44.53	1.62
	(60,40,20)	0.67	0.53	**42.27**	2.46
KPCA	(20,40,60)	0.64	**0.40**	33.62	**2.93**
	(40,40,40)	0.57	0.87	30.70	3.63
	(60,40,20)	**0.44**	1.78	**28.53**	5.67
KECA	(20,40,60)	0.66	**0.39**	33.38	**2.94**
	(40,40,40)	0.59	0.87	30.89	3.61
	(60,40,20)	**0.54**	1.87	**28.09**	5.75
Image Fig.1-c	(L_3,L_5,L_7)	α_C	β_C	α_B	β_B
OSVM_$\mathcal{X}^m$		0.97	**0.11**	58.87	**1.79**
OSVM	(20,40,60)	**0.77**	**2.12**	33.25	**7.32**
	(40,40,40)	0.80	3.28	29.99	9.01
	(60,40,20)	0.85	5.20	**28.11**	11.71
KPCA	(20,40,60)	0.65	**4.16**	23.27	**11.13**
	(40,40,40)	**0.61**	5.57	22.24	12.38
	(60,40,20)	0.62	8.78	**21.51**	15.89
KECA	(20,40,60)	0.65	**4.40**	22.69	**11.24**
	(40,40,40)	**0.58**	6.60	20.82	13.86
	(60,40,20)	0.60	10.78	**19.71**	18.19

Table 2: The error rate (%) with $\alpha = 0.05$

Image Fig.1-a	(L_3,L_5,L_7)	α_C	β_C	α_B	β_B
OSVM_$\mathcal{X}^m$		4.03	**0.00**	82.12	**0.30**
OSVM	(20,40,60)	**3.84**	**0.004**	60.81	**0.47**
	(40,40,40)	3.92	0.008	56.93	0.56
	(60,40,20)	4.22	0.01	**54.24**	0.67
KPCA	(20,40,60)	4.23	**0.01**	35.85	**2.62**
	(40,40,40)	4.04	0.07	32.56	2.98
	(60,40,20)	**3.95**	0.14	**31.95**	3.18
KECA	(20,40,60)	4.17	**0.004**	37.08	**2.41**
	(40,40,40)	4.11	0.05	33.31	2.65
	(60,40,20)	**3.86**	0.19	**30.34**	3.69
Image Fig.1-b	(L_3,L_5,L_7)	α_C	β_C	α_B	β_B
OSVM_$\mathcal{X}^m$		4.60	**0.00**	78.21	**0.27**
OSVM	(20,40,60)	**3.93**	**0.008**	60.61	**0.48**
	(40,40,40)	4.22	0.02	58.88	0.55
	(60,40,20)	4.17	0.05	**57.33**	0.77
KPCA	(20,40,60)	4.19	**0.03**	53.08	**0.81**
	(40,40,40)	3.65	0.05	49.49	1.07
	(60,40,20)	**3.35**	0.20	**46.00**	1.73
KECA	(20,40,60)	4.28	**0.03**	52.59	**0.82**
	(40,40,40)	3.68	0.07	48.64	1.10
	(60,40,20)	**3.52**	0.27	**45.19**	1.86
Image Fig.1-c	(L_3,L_5,L_7)	α_C	β_C	α_B	β_B
OSVM_$\mathcal{X}^m$		5.62	**0.04**	72.07	**0.49**
OSVM	(20,40,60)	4.85	**0.19**	55.72	**1.65**
	(40,40,40)	**4.77**	0.28	52.48	2.06
	(60,40,20)	4.83	0.57	**49.85**	3.04
KPCA	(20,40,60)	4.61	**0.41**	46.54	**2.81**
	(40,40,40)	**4.48**	0.86	41.31	3.99
	(60,40,20)	4.62	1.77	**37.87**	5.64
KECA	(20,40,60)	4.62	**0.44**	45.32	**2.92**
	(40,40,40)	**4.56**	0.94	40.30	4.23
	(60,40,20)	4.64	2.04	**36.35**	6.15

6. REFERENCES

[1] Ho Tin Kam, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 832–844, August 1998.

[2] X. He, P. Beauseroy, and A. Smolarz, "Random feature subset selection in a nonstationary environment : Application to textured image segmentation," *Proceedings of the 15th IEEE International Conference on Image Processing*, pp. 3028–3031, 2008.

[3] Leo Breiman, "Random forests," in *Machine Learning*, 2001, vol. 45, pp. 5–32.

[4] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, pp. 1443–1471, 2001.

[5] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, pp. 1299–1319, 1998.

[6] H. Hoffmann, "Kernel pca for novelty detection," *Pattern Recogn.*, vol. 40, pp. 863–874, 2007.

[7] R. Jenssen, "Kernel entropy component analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 847–860, 2010.

[8] Jolliffe I.T., *Principal Component Analysis*, Springer Verlag, 1986.

[9] A. Renyi, "On measures of entropy and information," *Selected Papers of Alfred Renyi*, vol. 2, pp. 565–580, 1976.

[10] K.I. Diamantaras and S.Y. Kung, "Principal component neural networks : Theory and applications," 1996.

[11] P. Brodatz, *Textures : A Photographic Album for Artists and Designers*, Dover Publication, 1966.

[12] A. Smolarz, "Etude qualitative du modèle autobinomial appliqué à la synthèse de texture," *XXIXèmes journées de Statistique*, pp. 712–715, 1997.