PCA Gaussianization for One-Class Remote Sensing Image Classification

Valero Laparra, Jordi Muñoz-Marí, Gustavo Camps-Valls and Jesús Malo

Image Processing Laboratory (IPL), Universitat de València
Catedrático A. Escardino - 46980 Paterna, València, Spain
{lapeva,jordi,gcamps,jmalo}@uv.es

ABSTRACT

The most successful one-class classification methods are discriminative approaches aimed at separating the class of interest from the outliers in a proper feature space. For instance, the support vector domain description (SVDD) has been successfully introduced for solving one-class remote sensing classification problems when scarce and uncertain labeled data is available. The success of this kernel method is due to that maximum margin nonlinear separation boundaries are implicitly defined, thus avoiding the hard and ill-conditioned problem of estimating probability density functions (PDFs). Certainly, PDF estimation is not an easy task, particularly in the case of high-dimensional PDFs such as is the case of remote sensing data. In high-dimensional PDF estimation, linear models assumed by widely used transforms are often quite restrictive to describe the PDF. As a result, additional non-linear processing is typically needed to overcome the limitations of the models. In this work we focus on the *multivariate Gaussianization* method for PDF estimation. The method is based on the *Projection Pursuit Density Estimation* (PPDE) technique.[1] The original PPDE procedure consists in iteratively project the data in the most non-Gaussian directions (like in ICA algorithms) and Gaussianizing them marginally. However, the extremely high computational cost associated to multiple ICA evaluations has prevented its practical use in high-dimensional problems such as those encountered in image processing. Here, we propose a fast alternative to iterative Gaussianization that makes it suitable for remote sensing applications while ensuring its theoretical convergence. Method's performance is successfully illustrated in the challenging problem of urban monitoring.

Keywords: Projection Pursuit, Gaussianization, PCA, density estimation, one-class, image classification, urban monitoring

1. INTRODUCTION

During the last decade, many methods have been developed to classify remote sensing images. The field comprises different learning paradigms, either supervised, unsupervised or semi-supervised. In the case of supervised classification, the user is given a number of labeled pixels belonging to different classes to develop a model that extrapolates well to unseen situations. The image classification problem is complex due to the potentially high dimension of available samples, low-sized labeled datasets, the presence of different noise sources, the non-stationary behaviour of land-cover spectral signatures, and the nonlinearities involved in the problem.[2] In such difficult situations, classifiers should produce accurate land-cover maps.

However, in many remote sensing applications, acquiring ground truth information for all classes is very difficult, especially when complex and heterogeneous geographical areas are analyzed. Actually, many other applications have turned to recognize one specific land-cover class of interest and to discriminate it from the other classes present in the investigated area. This formulation of the problem relaxes the constraint of having an exhaustive training set, but requires the availability of representative training data for the analyzed class and, if possible, some training samples representative of other classes, considered as outliers. Lately, high interest has been payed to this approach through the fields of: 1) *anomaly detection*, where one tries to identify pixels differing significantly from the background; and 2) *target detection*, where the target spectral signature is assumed to be known (or available from spectral libraries), and the goal is to detect pixels that match the target; and 3)

This work was partially supported by projects CICYT-FEDER TEC2006-13845, AYA2008-05965-C04-03 and CSD2007-00018, and grant BES2007-16125.

one-class classification, where one tries to detect one class and reject the others. In this paper, we focus on this latter problem of one-class classification.

Classical one-class classification methods are based on single hypothesis testing methods, which are intrinsically devoted to recognize the samples of one specific class from a heterogeneous distribution. Examples include the one-class Gaussian and the mixture of Gaussians methods, where the target class is either modeled using just one Gaussian or a mixture of K Gaussians, respectively.[3] The methods may work well in some well-sampled scenarios but the general assumption of Gaussianity in the input domain is not certainly a good choice in many cases. Another popular one-class method is the k-nearest neighbor algorithm, in which labels to test samples are assigned by computing the kth normalized distance to their nearest neighbor.[4] Despite its good performance in general, the accuracy of the k-nearest neighbor decreases severely if not enough training data is available, the data is composed of heterogeneous features or modalities, or when dealing with high dimensional feature spaces.[5] These are the common situations in remote sensing data classification: typically low number of training samples are available, one is usually interested in combining multisource information which dramatically increases data dimensionality. Despite these shortcomings, some applications of these methods can be found for remote sensing applications.[6–9] Lately, the introduction of kernel methods[10,11] alleviated these problems: in particular, the support vector domain description (SVDD) method has been recently introduced for specific one-class classification problems in remote sensing.[7,12–16] The success of kernel methods in general, and of the SVDD in particular, is due to that maximum margin nonlinear separation boundaries are defined implicitely, thus avoiding the hard and ill-conditioned problem of estimating probability density functions (PDFs).

In this paper, we focus on tackling the high-dimensional PDF estimation problem. For this purpose, we propose a simple method based on transforming the labeled image data to a statistically tractable feature space: the multivariate Gaussian. This, of course, cannot be done with a single linear transform due to its intrinsic limitations. For instance, PCA and local DCT assume a Gaussian source, while linear ICA and wavelets assume that images come from the *linear* combination of independent sources. These assumptions are not completely correct: for instance, a usual combination rule in natural scenes such as occlusion is intrinsically non-linear. This implies that residual relations among features still remain after any linear transform. The unsuitability of linear transforms to encompass the complexity of natural images implies that a number of tricks have to be added after the linear transform in order to describe the remaining relations. Examples of successful characterization of post-transforms relations include texture synthesis,[17] image coding,[18,19] or image denoising.[20]

On the contrary, the class of techniques collectively known as *projection pursuit*[1,21] may be applied to very general PDFs. Using projection pursuit for hyperspectral image classification has been studied previously in 22–24. Jimenez and Landgrebe[22] designed a projection index based on Bhattacharyya's distance to reduce the dimensionality of feature space. Ifarragaerri and Chang[23] used the information divergence (relative entropy) criterion to look for interesting projections that deviate from Gaussian distributions. Chiang and co-workers[24] developed evolutionary algorithms to find the best linear transform for a number of projection indices. However, none of the above approaches used projection pursuit to deal with the general PDF estimation problem.

Projection pursuit density estimation techniques solve the high-dimensional estimation problem by successive marginal univariate solutions thus circumventing the curse of dimensionality. For instance, the Gaussianization procedure proposed in Ref. 25 performs a series of linear ICA transforms followed by marginal Gaussianization in every transformed dimension. We will refer to this particular projection pursuit technique as G-ICA: Gaussianization through iterative ICA and marginal Gaussianization. Since convergence is guaranteed, after an *appropriate* number of iterations, any arbitrary PDF can be turned into a unit variance multidimensional Gaussian, and thus (unlike linear transforms) complete independence among features is achieved. The richness of the PDF under consideration is captured by the series of ICA transforms and the corresponding marginal non-linearities.

The weakness of general projection pursuit density estimation techniques, and also of G-ICA, is their computational cost. Note that, in this case, ICA is performed in each iteration: robust ICA algorithms such as RADICAL[26] lead to extremely slow convergence while convenient alternatives such as FastICA[27] may not converge. This explains why, so far, G-ICA has been applied just to low-dimensional signals.[28,29] These problems could be alleviated by the recently proposed single-step (non-iterative) Gaussianization transforms.[30,31] Unfortunately, these single step procedures are *also* restricted to particular PDF classes: (1) PDFs defined in convex

domains so that the final Gaussian can be achieved by marginal Gaussianization of every dimension in the appropriate axes,[30] or (2) elliptically symmetric PDFs so that the final Gaussian can be achieved by equalizing the length (norm) of the whitened samples.[31] In the case of images, the elliptical symmetry, and consequently convex domain, is true for small image patches (e.g. 10×10 pixels), but does not hold for bigger neighborhoods.[31] Moreover, the PDF of the spectral signatures may not necessarily fulfill these constraints. According to this, a general (yet computationally affordable) PDF estimation technique suited to image processing applications is not available yet.

In this work, we propose a fast alternative to G-ICA[25] that makes it suitable for high-dimensional remote sensing image applications. In each iteration, we use the standard PCA as an alternative to linear ICA, thus obtaining the desired result through iterated marginal Gaussianization and PCA (G-PCA). We show that using orthogonal linear transforms in the procedure does not change the theoretical convergence nor the convergence rate in practice. As a result, the proposed method reduces computation time by more than one order of magnitude, while keeping the appealing properties of the original method.

The paper is outlined as follows. In Section 2 we present the G-PCA transform compared to the previous G-ICA technique. Then we prove its theoretical convergence, and show that, in practice, G-PCA converges to G-ICA-like solutions in a fraction of the time. Section 3 analyzes the Jacobian of the G-ICA transform, which is the key factor for PDF estimation. The ability of G-PCA for PDF estimation is illustrated by using 2D examples. Section 4 shows the experimental results of the proposed method in non-linearly separable classification problems and in the challenging multispectral and multisource urban monitoring. Finally, Section 5 draws the conclusions of the work.

2. PCA GAUSSIANIZATION (G-PCA)

The inspiring paper of Chen and Gopinath[25] proposed a multivariate Gaussianization technique to turn any random variable into a unit variance multidimensional Gaussian by recursive application of linear ICA and marginal Gaussianization of the transformed variables. In this particular projection pursuit method, the general idea of seeking for *interesting* projections reduces to looking for the most independent projected features (linear ICA features) in each iteration. Beyond the theoretical convergence to a Gaussian, the nice property of G-ICA is that the transform is *invertible* and *differentiable*. Invertibility allows us to achieve solutions in the original domain while operating in a well-characterized (Gaussian) domain. Differentiability, allows to estimate the PDF in the original domain from the Jacobian of the transform in each point. However, as stated above, reliable ICA algorithms are extremely slow while fast algorithms may not converge.

Here, we propose to solve the aforementioned problems of G-ICA replacing linear ICA transforms with a series of orthogonal transforms obtained through linear PCA. Accordingly, we will refer to this PCA-based Gaussianization as G-PCA. Unlike ICA, using PCA ensures a closed form stable and unique solution in each iteration while dramatically reducing the computational burden. We, in addition, exchange the order of marginal Gaussianization and linear transform in each iteration. Even though this does not induce qualitative differences for a sufficiently large number of iterations, it is mathematically more convenient to prove method's convergence.

The proposed algorithm is summarized as follows: given a d-dimensional random variable $\mathbf{x}^{(0)} = [x_1, \ldots, x_d]^\top$, following a PDF, $p(\mathbf{x}^{(0)})$, in each iteration k, G-PCA performs:

$$\mathbf{x}^{(k+1)} = \mathbf{B}_{(k)} \cdot \Psi_{(k)}(\mathbf{x}^{(k)}), \tag{1}$$

where $\Psi_{(k)}$ is the marginal Gaussianization of each dimension of $\mathbf{x}^{(k)}$ for the corresponding iteration, and $\mathbf{B}_{(k)}$ is the PCA transform matrix for the marginally Gaussianized variable $\Psi_{(k)}(\mathbf{x}^{(k)})$. Marginal Gaussianization in each dimension, $\Psi^i_{(k)}$, can be decomposed into two equalization transforms: (1) marginal uniformization, $U^i_{(k)}$, based on the cumulative density function of the marginal PDF, and (2) Gaussianization of a uniform variable, $G(u)$, based on the inverse of the cumulative density function of a univariate Gaussian: $\Psi^i_{(k)} = G \odot U^i_{(k)}$, where:

$$u = U^i_{(k)}(x^{(k)}_i) = \int_{-\infty}^{x^{(k)}_i} p_i(x'^{(k)}_i)\, dx'^{(k)}_i \tag{2}$$

$$G^{-1}(x_i) = \int_{-\infty}^{x_i} g(x'_i)\, dx'_i \tag{3}$$

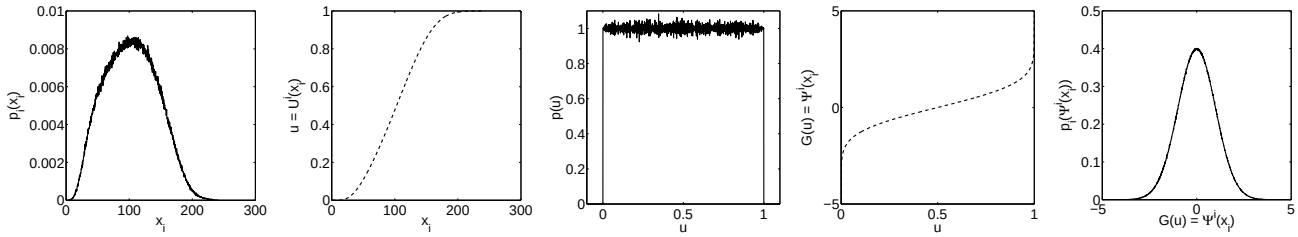

Figure 1. Example of marginal Gaussianization in a particular dimension i. From left to right: marginal PDF of x_i, uniformization transform $u = U^i(x_i)$, PDF of the uniformized variable $p(u)$, Gaussianization transform $G(u)$, and PDF of the Gaussianized variable $p_i(\Psi^i(x_i))$.

and $g(x_i)$ is just a univariate Gaussian. Figure 1 shows an example of the marginal Gaussianization of a one-dimensional variable x_i (Landsat TM channel 3 of the Naples scene used in the experiments, cf. Section 5.2).

While the proposed modifications of the original method may seem naïve, some non-trivial questions arise:

- Is convergence of the new algorithm guaranteed?
- Do G-ICA and G-PCA solutions differ?

In the following subsections we address these questions both theoretically and experimentally.

2.1 Theoretical convergence of G-PCA

In order to prove the convergence of G-PCA we have to show that the Kullback-Leibler (KL) divergence between the PDF of the transformed random variable and a multivariate Gaussian (the negentropy, J) is reduced in each iteration k. If $\Delta J^{(k)} \geq 0 \; \forall \, k$, the transformed variable asymptotically tends to a Gaussian.

The negentropy of a random variable, $\mathbf{x}$, may be expressed as:[31]

$$J(\mathbf{x}) = I(\mathbf{x}) + \sum_{i=1}^{d} J_M(x_i) - \underbrace{\left(\sum_{i=1}^{d} \log(\Sigma_{ii}) - \log|\mathbf{\Sigma}| \right)}_{2\text{nd ord}(\mathbf{x})} \tag{4}$$

where, $I(\mathbf{x})$ is the multi-information among the coefficients (features) of the random variable $\mathbf{x} \in \mathbb{R}^d$, $J_M(x_i)$ is the marginal negentropy (i.e. the divergence between the marginal PDF of x_i and a univariate Gaussian), and the last term, related to the covariance matrix, $\mathbf{\Sigma}$, describes the second order relations between the features of the random variable.

The negentropy reduction in each iteration, k, of G-PCA is:

$$\begin{aligned} \Delta J^{(k)} &= J(\mathbf{x}^{(k)}) - J(\mathbf{x}^{(k+1)}) \\ &= J(\mathbf{x}^{(k)}) - J(\mathbf{B}_{(k)} \cdot \Psi_{(k)}(\mathbf{x}^{(k)})) \end{aligned} \tag{5}$$

The negentropy is invariant under orthogonal transforms, $\mathbf{B}_{(k)}$, because these imply pure rotations and hence the divergence between the rotated PDF and the (spherically symmetric) Gaussian is the same. Therefore, by applying Eq. (4), one can readily express the negentropy reduction as follows:

$$\begin{aligned} \Delta J^{(k)} &= J(\mathbf{x}^{(k)}) - J(\Psi_{(k)}(\mathbf{x}^{(k)})) \\ &= I(\mathbf{x}^{(k)}) + \sum_{i=1}^{d} J_M(x_i^{(k)}) - 2\text{nd ord}(\mathbf{x}^{(k)}) - I(\Psi_{(k)}(\mathbf{x}^{(k)})) - \sum_{i=1}^{d} J_M(\Psi_{(k)}(\mathbf{x}^{(k)})_i) + 2\text{nd ord}(\Psi_{(k)}(\mathbf{x}^{(k)})) \end{aligned} \tag{6}$$

Taking into account that multi-information is invariant under dimension-wise transforms, $I(\mathbf{x}^{(k)}) = I(\Psi_{(k)}(\mathbf{x}^{(k)}))$; given that marginally Gaussianized variables have zero marginal negentropy, $\sum_{i=1}^{d} J_M(\Psi_{(k)}(\mathbf{x}^{(k)})_i) = 0$; and

considering that the second order relations in $\mathbf{x}^{(k)}$ are removed by the PCA in the previous iteration, $\mathbf{B}_{(k-1)}$; we have:

$$\Delta J^{(k)} = \sum_{i=1}^{d} J_M(x_i^{(k)}) + 2\text{nd ord}(\Psi_{(k)}(\mathbf{x}^{(k)})) \geq 0 \tag{7}$$

Since marginal negentropies and second order relations are always equal or bigger than zero, we proved that negentropy is reduced in each iteration, thus ensuring convergence to a multivariate Gaussian.

The previous theoretical convergence limit provides a practical criterion to stop the iterative process. Note that one should stop the series of transforms when the reduction in negentropy (distance to a Gaussian) is small enough.

2.2 Convergence of G-PCA in practice

Here, we experimentally analyze two important characteristics of G-PCA: the convergence rate and the computational cost. Figure 2 illustrates the performance of our method in a 2D highly non-Gaussian manifold compared to the G-ICA result. In this case we used the FastICA algorithm[27] to speed up G-ICA. The proposed early-stopping criterion was applied. Figure 2 shows how at each iteration the (accumulated) multi-information reduction, ΔI, converges to a constant value for both G-ICA and G-PCA, and our method achieves virtually the same results with a slightly higher number of iterations ($N = 37$ vs $N = 25$). Note that there is a direct functional relation between the negentropy reduction ΔJ and the mutual-information reduction ΔI (equation 4). Although G-PCA converges after more iterations, this does not imply a higher computational load, as PCA is much cheaper than ICA. This advantage is more relevant in higher dimensional problems. To assess this, we Gaussianized patches of different sizes from the standard grayscale image 'Barbara'. Results for both CPU time and the achieved ΔI are presented in Table 1. For similar ΔI reductions, more than one order of magnitude in computation time is gained by G-PCA, e.g. when working with 64 dimensions (8×8 patches), G-PCA takes about 4 minutes while G-ICA takes around 4 hours.

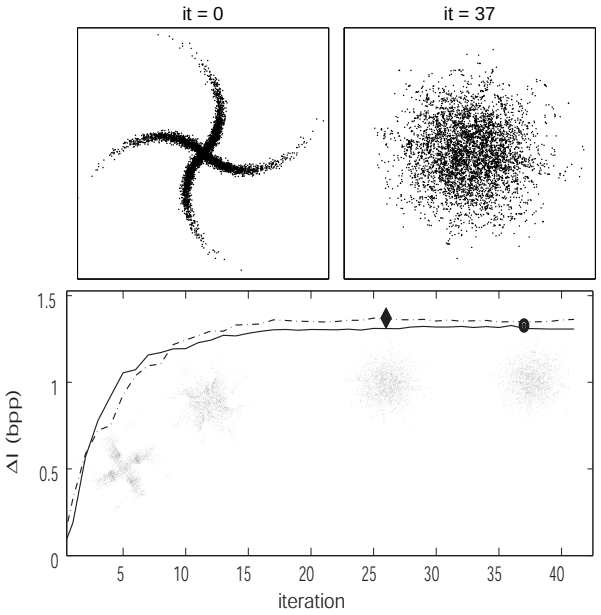

Figure 2. Performance of G-PCA in a toy example. Original and transformed data (top), and cumulative ΔI for each iteration for G-PCA (solid) and G-ICA (dashed). Optimal iterations are highlighted. Inset scatter plots show the achieved G-PCA solution at different iterations.

Table 1. Cumulative ΔI and CPU time for G-ICA and G-PCA.

	G-ICA		**G-PCA**	
dim	ΔI [bpp]	Time [s]	ΔI [bpp]	Time [s]
2×2	1.54	865	1.51	14
3×3	2.08	1236	2.05	34
4×4	2.38	2197	2.29	63
5×5	2.50	3727	2.44	99
6×6	2.60	6106	2.56	141
7×7	2.68	9329	2.63	170
8×8	2.69	15085	2.69	233

3. PDF ESTIMATION WITH G-PCA

Given two random variables $\mathbf{x}$ and $\mathbf{y}$, such as $\mathbf{y} = \mathcal{G}(\mathbf{x})$, the PDF in the input domain, $p_{\mathbf{x}}(\mathbf{x})$, is related to the PDF in the transform domain, $p_{\mathbf{y}}(\mathbf{y})$, according to Ref. 32:

$$p_{\mathbf{x}}(\mathbf{x}) = p_{\mathbf{y}}(\mathcal{G}(\mathbf{x}))\,|\nabla_{\mathbf{x}}\mathcal{G}| \tag{8}$$

In the G-PCA case the PDF in the transformed domain is a multivariate unit variance Gaussian, so the only unknown in the above equation is the Jacobian of the G-PCA transform, $\nabla_{\mathbf{x}}\mathcal{G}$.

The Jacobian of the series of N transforms is the product of the Jacobians in each iteration:

$$\nabla_{\mathbf{x}}\mathcal{G} = \prod_{k=1}^{N} \mathbf{B}_{(k)} \cdot \nabla_{\mathbf{x}^{(k)}} \Psi_{(k)} \tag{9}$$

Marginal Gaussianization, $\Psi_{(k)}$, is a dimension-wise transform, whose Jacobian is the diagonal matrix,

$$\nabla_{\mathbf{x}^{(k)}} \Psi_{(k)} = \begin{pmatrix} \dfrac{\partial \Psi^1_{(k)}}{\partial x^{(k)}_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \dfrac{\partial \Psi^d_{(k)}}{\partial x^{(k)}_d} \end{pmatrix} \tag{10}$$

According to the two equalization steps in each marginal Gaussianization (Eq. 3), each element in $\nabla_{\mathbf{x}^{(k)}} \Psi_{(k)}$ is:

$$\frac{\partial \Psi^i_{(k)}}{\partial x^{(k)}_i} = \frac{\partial G}{\partial u} \cdot \frac{\partial u}{\partial x^{(k)}_i} = \left(\frac{\partial G^{-1}}{\partial x_i} \right)^{-1} \cdot p_i(x^{(k)}_i) = g(\Psi^i_{(k)}(x^{(k)}_i))^{-1} \cdot p_i(x^{(k)}_i) \tag{11}$$

It is important to note that the proposed multivariate PDF estimation using G-PCA and expressions (8)-(11) just depends on univariate (marginal) PDF estimations. Therefore the proposed method does not suffer from the curse of dimensionality.

Figure 3 shows a 2D example of PDF estimation using G-PCA and the above expressions. In this case, 10^4 samples were used in the G-PCA transform and in the histogram. The PDF was estimated in 50×50 points of the domain. The same resolution in bins was used to compute the histogram. Note that the G-PCA estimation of the PDF is much smoother than naïve estimation using a simple histogram.

4. RELATION OF G-PCA TO OTHER METHODS

In this section, we point out some particularly interesting relations of the proposed G-PCA to the kernel-based SVDD and artificial neural networks.

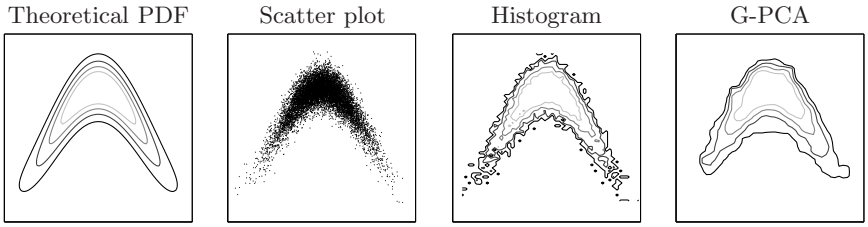

Figure 3. Example of PDF estimation. From left to right: theoretical PDF, scatter plot of the data used in the estimations, histogram estimation using a number of bins to obtain the same resolution as in the G-PCA estimation.

4.1 Relation to the Support Vector Domain Description (SVDD)

The proposed G-PCA and the Support Vector Domain Description (SVDD) method may be seen as conceptually similar due to their *apparent* geometrical similarity. However, G-PCA and SVDD represent two different approaches to the classification problem: PDF estimation versus separation boundary estimation. On the one hand, SVDD finds a minimum volume sphere in a kernel feature space that contains $1-\nu$ fraction of the *target* training samples.[33] On the other hand, naive interpretation of G-PCA may be seen as if test samples were transformed and classified as *target* if lying inside the sphere containing $1-\nu$ fraction of the learned Gaussian distribution. According to this interpretation, both methods reduce to computing spherical boundaries in different feature spaces. However, this is not completely true in the G-PCA case: note that the value of the G-PCA Jacobian is not the same at every location in the Gaussianized domain. Therefore, the optimal boundary to reject a ν fraction of the training data is not necessarily a sphere in the Gaussianized domain. In the case of the SVDD, though, by using an isotropic RBF kernel, all directions in the kernel feature spaces are treated in the same way.

Strictly speaking, the solution of classification problems does not need PDF estimation but boundary estimation. This dilemma cuts to the heart of that pointed out by Vapnik [34, pg. 30]: *"When solving a given problem, try to avoid solving a more general problem as an intermediate step"*. This rationale suggests not tackling the general problem of density estimation for classification but rather the more specific, tractable and direct problem of large margin separation. However, the techniques looking for the optimal classification boundary, such as SVDD, may need non-target labeled samples *all around* the class of interest in order to find a proper set of free parameters allowing to learn the support of the target class. The lack of non-target samples could lead to incorrect boundary estimates, as it is difficult, and for some problems even impossible, to find a proper set of free parameters. This issue becomes critical in high dimensional problems since the boundary surface increases with dimensionality. On the contrary, the rejection fraction (and the associated classification boundary) in G-PCA is set according to the probability of the target class, so it only depends on the amount of target class samples. This is particularly important in one-class classification problems since the target class may be well characterized, while accurate characterization of all other possible classes is neither generally available nor actually needed. This problem will be illustrated in the results section.

4.2 Relation to deep neural networks

The proposed Gaussianization method is essentially a sequence of two operations: a linear transform followed by a non-linear squashing function. This processing is intuitively very similar to that carried out in a feedforward neural network (linear transform plus sigmoid-shaped function in each hidden layer). Therefore, one could see each iteration of the G-PCA as one hidden layer processing of the data, and thus argue that complex (highly non-Gaussian) tasks should require more hidden layers (iterations). This view is in line with the field of *deep learning* in neural networks, which consists of learning a model with several layers of nonlinear mappings. The field is very active nowadays because some tasks, such as natural language processing or speech recognition, are highly nonlinear. Note, that it may appear counterintuitive the fact that full Gaussianization of a dataset is eventually achieved with a large enough number of iterations, thus leading to overfitting in the case of a neural network with such number of layers. Nevertheless, note that capacity control also applies here: we have observed that early-stopping criteria must be strictly applied to allow good generalization properties. In this setting, one can see early stopping in G-PCA as a form of model regularization. This is certainly an interesting research line to be pursued in the future.

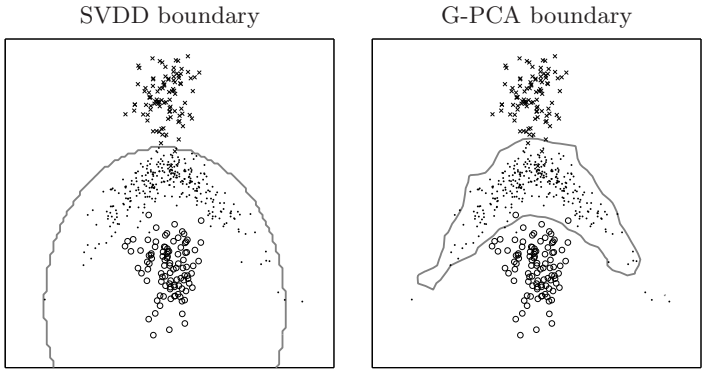

Figure 4. Outlier detection in a non-linearly separable 2D problem. The target class is represented by dots. Circles and crosses represent outliers of different nature. The figures show the classification boundaries found by SVDD (left) and G-PCA (right) when trained using a restricted set of outliers (crosses).

5. EXPERIMENTAL RESULTS

The proposed method is illustrated in two experiments and will be compared to standard SVDD because of their intuitive similarity (cf. Section 4.1). The first 2D experiment on synthetic data illustrates the capabilities of the method in a non-linearly separable and poorly sampled one-class problem. The second experiment deals with real multispectral and multisource data and illustrates the advantages of G-PCA in real and challenging scenarios.

5.1 Experiment 1: 2D non-linearly separable problems.

In this 2D experiment the problem is detecting outliers from the target class represented by dots in Fig. 4. Two possible outlier classes were considered, represented by circles 'o' and crosses '×'. Imagine that, at the training stage, only outliers of the cross class are available. Then, the G-PCA transform, the width of the RBF kernel in SVDD, and the rejection ratios are trained to maximize the κ statistic[35] by using the available samples of the target class and the '×' class (10^3 and 10^2 samples, respectively). The classification boundaries for both methods and the referred training are represented in gray lines.

Note that the proposed method is able to identify/reject outliers of different nature ('o') while the SVDD solution is unable to discriminate these new outliers. This example stresses the advantages of PDF estimation of the target class in scenarios where all possible outliers are not available, or the space is not correctly sampled.

5.2 Experiment 2: Multisource one-class image classification

In this experiment, we assess the performance of the G-PCA classifier to detect urban areas from multispectral and SAR images. The images used in this section were collected in the Urban Expansion Monitoring (UrbEx) ESA-ESRIN DUP project.[36] For further details, visit http://dup.esrin.esa.int/ionia/projects/summaryp30.asp. The considered test sites were the cities of Rome and Naples, Italy, for two acquisition dates (1995 and 1999). The available features were the seven Landsat bands, two SAR backscattering intensities (0–35 days), and the SAR interferometric coherence. Since these features come from different sensors, the first step was to perform a specific processing and conditioning of optical and SAR data, and to co-register all images. We used all seven Landsat TM spectral bands (containing three *VIS*, one *Near IR*, two *Short-Wave IR*, and one *Thermal IR* bands). In the case of the SAR images, we also used a spatial version of the coherence specially designed to increase the urban areas discrimination.[36] After this preprocessing, all features were stacked at a pixel level, and each feature was standardized.

We compared the G-PCA classifier based on the estimated PDF for urban areas with the classifier based on the SVDD. We used the RBF kernel for the SVDD whose width was varied in the range $\sigma \in [10^{-2}, \ldots, 10^2]$. The fraction rejection parameter was varied in $\nu \in [10^{-2}, 0.5]$ for both methods. The optimal (best κ) parameters were selected through 3-fold cross-validation in the training set. Training sets of different size for the target class were used in the range $[100, 2500]$. We assumed a scarce knowledge of the non-target class: 10 outlier examples

were used in all cases. The test set was constituted by 10^5 pixels of each considered image. Training and test samples were randomly taken from the whole spatial extent of each image. The experiment was repeated for 10 different random realizations in the three images.

Figure 5 shows the κ statistic achieved by SVDD and G-PCA in the test set of the three considered images.The κ values are relatively small because samples were taken from a large spatial area thus giving rise to a challenging problem due to the variance of the spectral signatures. Results show that SVDD outperforms the proposed method for small size training sets. This is because more target samples are needed by the G-PCA for an accurate PDF estimation. However, for moderate and large training sets the proposed method substantially outperforms SVDD. Note that training size requirements of G-PCA are not too demanding: 750 samples on a 10-dimensional problem are enough for G-PCA to outperform SVDD when very little is known of the non-target class.

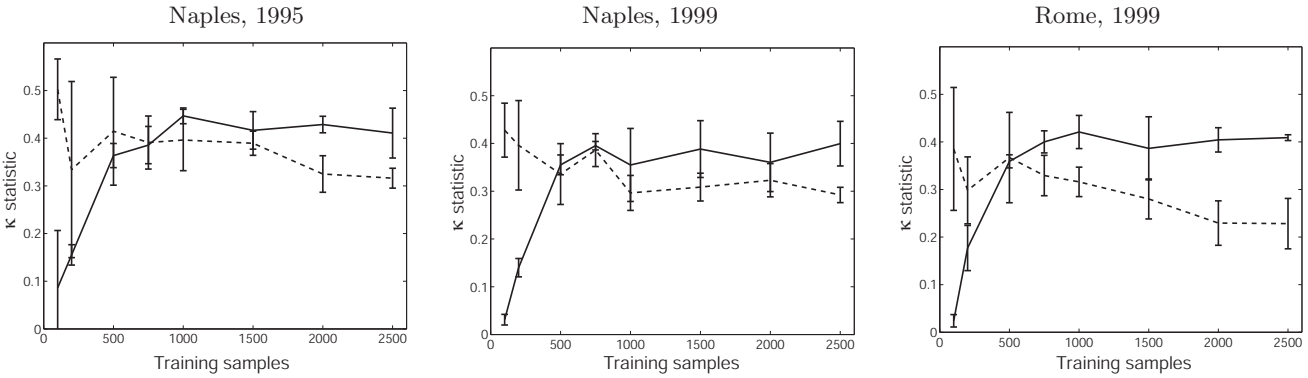

Figure 5. Classification performance (κ statistics) as a function of the number of training samples for the three considered images by the SVDD (dashed) and the G-PCA (solid).

.

Figure 6 shows the classification maps using a restricted training strategy. In this case, the experiment was carried out over a small region (200×200) of the Naples 1995 image. We used 2000 samples of the target class and only 10 samples of the non-target class. Here the classification performance (κ statistic) is better than the results reported in Fig. 5 because small regions have more homogeneous features, and then the variance of spectral signatures is smaller. As a consequence, the training data describes more accurately the particular behavior of the smaller spatial region thus achieving a better performance in the test set.

Note that, although the SVDD classification map is more homogeneous, G-PCA better rejects the 'non-urban' areas (in black). This may be because SVDD training with few non-target data gives rise to a too broad boundary as in the example of Fig. 4. As a result, too many pixels are identified as belonging to the target class (in white). Another relevant observation is the noise in neighboring pixels, which may come from the fact that no spatial information was used. This problem could be easily alleviated by imposing some post-classification smoothness constraint or by incorporating texture features for classification.

6. CONCLUSIONS

We proposed a fast alternative to iterative Gaussianization methods that makes it suitable in high-dimensional problems such as those in remote sensing applications. The proposed G-PCA consists of iteratively applying marginal Gaussianization and PCA to any original dataset. The result is a multivariate Gaussian. Theoretical convergence of the proposed method was proved.

The method exhibits fast and stable convergence rates through a suitable early-stopping criterion. The computational cost is dramatically reduced compared to ICA-based Gaussianization methods. The proposed Gaussianization technique can be used for accurate multivariate PDF estimation when a relatively small number of samples is available. This is because the proposed G-PCA reduces the multidimensional problem to a set of univariate (marginal) PDF estimation problems.

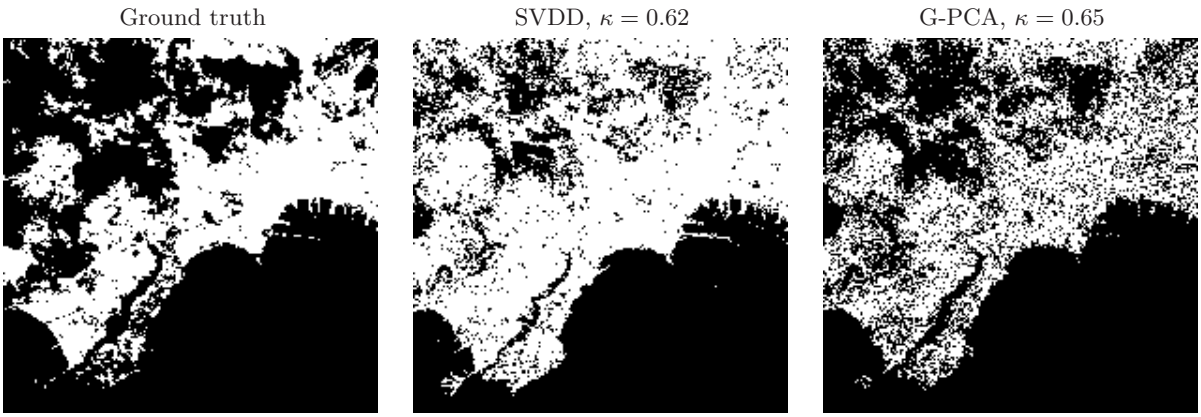

Figure 6. Classification performance over a small region of the Naples image (1995). White points represent urban areas while black points represent non-urban areas.

The experiments showed that G-PCA outperforms the (qualitatively similar) SVDD in realistic situations in which the target class is well known but not many examples of the non-target class are available. From an experimental viewpoint, our future work is tied to testing performance in hyperspectral image classification. In the theoretical side, a deeper analysis of the geometrical relationship with SVDD and deep neural networks will be carried out.

REFERENCES

[1] Huber, P. J., "Projection pursuit," *The Annals of Statistics* **13**(2), 435–475 (1985).

[2] Lillesand, T. M., Kiefer, R. W., and Chipman, J. W., [*Remote Sensing and Image Interpretation*], John Wiley, New York, 5th ed. (2004).

[3] Tax, D. M. J., *One-class classification: concept learning in the absence of counter-examples*, PhD thesis, Technische Universiteit Delft (2001).

[4] Shakhnarovich, G., Darrell, T., and Indyk, P., [*Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*], MIT Press, Cambridge, MA (Mar 2006).

[5] Beyer, k., Goldstein, J., Ramakrishnan, R., and Shaft, U., "When is 'nearest neighbor' meaningful?," *Lecture Notes in Computer Science* **1540**, 217–235 (2006).

[6] Carlotto, M., "A cluster-based approach for detecting man-made objects and changes in imagery," *IEEE Transactions on Geoscience and Remote Sensing* **43**, 374–387 (Feb. 2005).

[7] Muñoz-Marí, J., Bruzzone, L., and Camps-Valls, G., "A support vector domain description approach to supervised classification of remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing* **45**(8), 2683–2692 (2007).

[8] Samaniego, L., Bardossy, A., and Schulz, K., "Supervised classification of remotely sensed imagery using a modified k -nn technique," *IEEE Transactions on Geoscience and Remote Sensing* **46**, 2112–2125 (July 2008).

[9] Blanzieri, E. and Melgani, F., "Nearest neighbor classification of remote sensing images with the maximal margin principle," *IEEE Transactions on Geoscience and Remote Sensing* **46**, 1804–1811 (June 2008).

[10] Schölkopf, B. and Smola, A. J., [*Learning with Kernels*], MIT Press (2002).

[11] Shawe-Taylor, J. and Cristianini, N., [*Kernel Methods for Pattern Analysis*], Cambridge University Press (2004).

[12] Li, J. and Narayanan, R. M., "A shape-based approach to change detection of lakes using time series remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing* **41**, 2466–2477 (Nov 2003).

[13] Liu, D., Kelly, M., and Gong, P., "Classifying multi-temporal Landsat TM imagery using Markov random fields and support vector machines," in [*3rd International Workshop on the Analysis of Multi-temporal Remote Sensing Images*], (2005).

[14] Mercier, G. and Girard-Ardhuin, F., "Partially supervised oil-slick detection by SAR imagery using kernel expansion," *IEEE Transactions on Geoscience and Remote Sensing* **44**, 2839–2846 (Oct 2006).

[15] Potin, D., Vanheeghe, P., Duflos, E., and Davy, M., "An abrupt change detection algorithm for buried landmines localization," *IEEE Transactions on Geoscience and Remote Sensing* **44**, 260–272 (Feb 2006).

[16] Camps-Valls, G., Gómez-Chova, L., Muñoz-Marí, J., Martínez-Ramón, M., and Rojo-Álvarez, J. L., "Kernel-based framework for multi-temporal and multi-source remote sensing data classification and change detection," *IEEE Transactions on Geoscience and Remote Sensing* **46**, 1822–1835 (Jun 2008).

[17] Portilla, J. and Simoncelli, E. P., "A parametric texture model based on joint statistics of complex wavelet coefficients," *Int. J. Comp. Vis.* **40**(1), 49–71 (2000).

[18] Malo, J., Epifanio, I., Navarro, R., and Simoncelli, E., "Non-linear image representation for efficient perceptual coding," *IEEE Trans. Im. Proc.* **15**(1), 68–80 (2006).

[19] Camps-Valls, G., Gutiérrez, J., Gómez, G., and Malo, J., "On the suitable domain for SVM training in image coding," *Journal of Machine Learning Research* **9**, 49–66 (2008).

[20] Laparra, V., Gutiérrez, J., Camps-Valls, G., and Malo., J., "Recovering wavelet relations using SVM for image denoising," *IEEE Int. Conf. Im. Proc. 08* , 541–544 (2008).

[21] Friedman, J. and Tukey, J., "A projection pursuit algorithm for exploratory data analysis," *IEEE Trans. Comp.* **C-23**(9), 881–890 (1974).

[22] Jiménez, L. O. and Landgrebe, D. A., "Hyperspectral data analysis and supervised feature reduction via projection pursuit," *IEEE Transactions on Geoscience and Remote Sensing* **37**(6), 2653–2667 (1999).

[23] Ifarraguerri, A. and Chang, C., "Unsupervised hyperspectral image analysis with projection pursuit," *IEEE Transactions on Geoscience and Remote Sensing* **38**(6), 2529–2538 (2000).

[24] Chiang, S. and Chang, C., "Unsupervised target detection in hyperspectral images using projection pursuit," *IEEE Transactions on Geoscience and Remote Sensing* **39**(7), 1380–1391 (2001).

[25] Chen, S. and Gopinath, R., "Gaussianization," in [*NIPS*], 423–429 (2000).

[26] Learned, E. and Fisher, J., "ICA using spacings estimates of entropy," *J. Mach.Learn.Res.* **4**, 1271–1295 (2003).

[27] Hyvärinen, A., "Fast and robust fixed-point algorithms for ICA," *IEEE Trans. Neur. Net.* **10**, 626–634 (1999).

[28] Xiang, B., Chaudhari, U., Ramaswamy, G., and Gopinath, R., "Short-time gaussianization for robust speaker verification," in [*IEEE ICASSP 02'*], (2002).

[29] Zhang, K. and Chan, L., "Extended gaussianization method for blind separation of post-nonlinear mixtures," *Neur. Comp.* **17**(2), 425–452 (2005).

[30] Erdogmus, D., Jenssen, R., Rao, Y., and Principe, J., "Gaussianization: An efficient multivariate density estimation technique for statistical signal processing," *J. VLSI Sig. Proc.* **45(17)**, 67–83 (2006).

[31] Lyu, S. and Simoncelli, E. P., "Nonlinear extraction of 'independent components' of natural images using Radial Gaussianization," *Neur. Comp.* **21**(6), 1485–1519 (2009).

[32] Stark, H. and Woods, J., [*Probability, Random Processes and Estimation Theory for Engineers*], Prentice-Hall, Englewood Cliffs, New Jersey (1986).

[33] Tax, D. and Duin, R., "Support vector domain description," *Patt. Recogn. Lett.* **20**, 1191–1199 (1999).

[34] Vapnik, V. N., [*The Nature of Statistical Learning Theory*], Springer, New York, 2nd ed. (2000).

[35] Landis, J. and Koch, G. G., "The measurement of observer agreement for categorical data," *Biometrics* **33**(1), 159–174 (1977).

[36] Gómez-Chova, L., Fernández-Prieto, D., Calpe, J., Soria, E., Vila-Francés, J., and Camps-Valls, G., "Urban monitoring using multitemporal SAR and multispectral data," *Pattern Recognition Letters* **27**, 234–243 (Mar 2006). 3rd Pattern Recognition in Remote Sensing Workshop, Kingston Upon Thames, ENGLAND, AUG 27, 2004.