NON-NEGATIVE PRE-IMAGE IN MACHINE LEARNING FOR PATTERN RECOGNITION

Maya Kallas[(1,2)], *Paul Honeine*[(1)], *Cédric Richard*[(3)], *Clovis Francis*[(2)], *and Hassan Amoud*[(4)]

[(1)] Institut Charles Delaunay (CNRS), UMR, LM2S, Université de Technologie de Troyes, France
[(2)] Laboratoire d'analyse des systèmes (LASYS), Université Libanaise, Liban
[(3)] Laboratoire Fizeau (CNRS), Observatoire de la Côte d'Azur, Université de Nice Sophia-Antipolis, France
[(4)] Azm Center for Research in Biotechnology and its Applications, Lebanese University, Lebanon

ABSTRACT

Many real-life applications are nonlinear by nature. Moreover, in order to have a physical interpretation, some constraints should be incorporated in the signal or image processing technique, such as the non-negativity of the solution. This paper deals with the non-negative pre-image problem in kernel machines, for nonlinear pattern recognition. While kernel machines operate in a feature space, associated to the used kernel function, a pre-image technique is often required to map back features into the input space. We derive a gradient-based algorithm to solve the pre-image problem, and to guarantee the non-negativity of the solution. Its convergence speed is significantly improved due to a weighted stepsize approach. The relevance of the proposed method is demonstrated with experiments on real datasets, where only a couple of iterations are necessary.

1. INTRODUCTION

Constraints are often required to ensure the physical interpretation of many signal and image processing techniques. In pattern recognition on grayscale images, such as deconvolution, deblurring or denoising applications, the result should be a potential valid image. Commonly stored with 8 bits per sampled pixel, each pixel can have 2^8 values within the interval $[0, 255]$, defined by the weakest intensity (0 for black) and the strongest one (255 for white). As the intensity carries the information, numerous applications give rise to grayscale images with a small number of intense pixels, i.e., many pixels have null values. This is the case in many imaging problems, spanning areas such as biomedical and astrophysics [1]. For this reason, one often includes non-negativity constraint on the solution, which leads to zero-valued pixels. In early studies, the non-negativity constraint was introduced for signal deconvolution by Thomas in [2] and Prost *et al.* in [3]. Image deconvolution was studied by Thomas *et al.* in [4], while Snyder *et al.* introduced image deblurring in [5]. More recently, multiplicative algorithms for signal restoration were studied by Lantéri *et al.* in [1]. Most of these researches only focus on linear systems [6, 7], while many real-life applications exhibit nonlinear behavior.

Within the past decade or so, kernel-based machines have been increasingly used in machine learning [8]. Achieving high accuracy with low computational cost [9], they have been highly successful in solving many nonlinear problems in classification, regression, prediction and pattern recognition, only to name a few. They rely on the *kernel trick*, which transforms a linear algorithm into a nonlinear one as long as it can be expressed exclusively in terms of inner products between data. By using a positive semi-definite kernel instead of the inner product, we implicitly map the data from the input space into a feature space using a nonlinear map function. The kernel is called *reproducing kernel* and the corresponding feature space is the so-called *reproducing kernel Hilbert space* (RKHS).

Even though mapping to a feature space is important, it is usually more interesting to study patterns in the input space rather than their counterparts in the RKHS. However, both spaces are not in bijection and very few elements of the latter have a *pre-image* in the former. In general, the exact pre-image may not exist and, if it exists, it might not be unique. This defines the *pre-image problem*, as one seeks an element of the input space whose image, by the same kernel function, is as close as possible to some element in the feature space. Many techniques have been presented in literature in order to solve such nonlinear optimization problem. Mika *et al.* introduced this optimization problem and proposed a fixed-point iterative method in [10]. In [11], a multidimensional-scaling (MDS) method was introduced, while lately, in [12], a more direct method using relationship between inner-products was presented. See [13] for a recent review with several applications in signal processing.

In [14], we showed that the pre-image can be defined with a non-negative additivity of its contributions, by writing the pre-image as a linear model with the available data. However, the non-negativity of the pre-image has not been studied before. This paper deals with nonlinear pattern recognition under non-negativity constraint on the pre-image. To this end, we show that a gradient descent/ascent scheme can be prescribed for solving the pre-image problem, which is somewhat surprising due to the non-linearity and non-convexity of the optimization problem. By controlling its stepsize at each iteration, we guarantee the non-negativity of the solution. Moreover, by weighting the stepsize by the actual value, we derive an algorithm that converges faster, where the sparsity of the solution is privileged. A fortuitous side-effect of the proposed strategy is its re-

This work is partly supported under the French-Lebanese collaboration program CEDRE No. 10 SCI F15/L5.

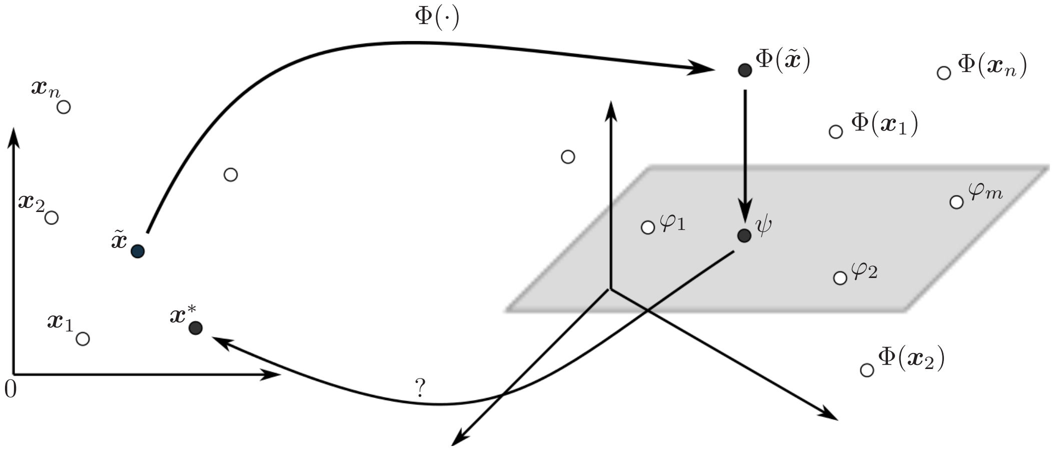


Figure 1: Schematic illustration of the pre-image problem for the denoising problem. A given noisy data $\tilde{\boldsymbol{x}}$ is mapped to $\Phi(\tilde{\boldsymbol{x}})$, then projected into the subspace spanned by the most relevant principal axis $\varphi_1, \varphi_2, \ldots, \varphi_m$. The denoised pattern ψ is mapped back into the input space, to $\boldsymbol{x}^*$.

markable self-regularization property. This is illustrated on an image denoising task, where only a couple of iterations were necessary. The proposed strategy outperforms conventional unconstrained pre-image techniques.

The rest of the paper is organized as follows: In the next section, we introduce kernel-based machines with the kernel PCA and describe the denoising scheme. In Section 3, we study the pre-image problem for pattern recognition, illustrated with a gradient approach, while in Section 4, we consider imposing non-negativity constraint on the pre-image. Section 5 gives experimental results illustrating the efficiency of the proposed method on real handwritten digits from the MNIST datasets.

2. KERNEL PCA FOR PATTERN RECOGNITION AND DENOISING

Let $\mathbb{R}^d$ be the input space to which we associate the Euclidean dot product $\boldsymbol{x}_i \cdot \boldsymbol{x}_j$ for any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^d$. Let $\kappa\colon \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}^d$ be a positive semi-definite kernel, that is $\sum_{i,j} \alpha_i \alpha_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$ for all $\alpha_i, \alpha_j \in \mathbb{R}$ and $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^d$. The Moore-Aronszajn theorem [9] states that every positive semi-definite kernel is associated with a unique reproducing kernel Hilbert space $\mathcal{H}$, and vice-versa. This statement is mathematically expressed by a mapping function, $\Phi\colon \mathbb{R}^d \mapsto \mathcal{H}$, from the input space into the feature space, such that $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}$, for any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^d$, where $\langle \cdot\,, \cdot \rangle_{\mathcal{H}}$ defines the inner product in $\mathcal{H}$.

Next, we study a particular kernel-based machine for nonlinear pattern recognition: kernel PCA, a nonlinear version of the mostly used Principal Component Analysis (PCA). The main idea in PCA is to extract the most relevant directions, and thus the pertinent subspace, from a collection of available data. These directions correspond to the eigenvectors with the highest eigenvalues of the correlation matrix. The same concept can be applied in the feature space, which give rise to the kernel PCA algorithm.

To this end, each data is mapped into a feature space with $\Phi\colon \mathbb{R}^d \mapsto \mathcal{H}$, where conventional PCA is applied. Let $\Phi(\boldsymbol{x}_1), \Phi(\boldsymbol{x}_2), \ldots, \Phi(\boldsymbol{x}_n) \in \mathcal{H}$ denote the mapped data. In practice, one does not require the explicit form of the map, since most computations can be done using the concept of the *kernel trick*, i.e., with $\langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}$, for $i, j = 1, 2, \ldots, n$. By analogy to PCA, each relevant axis φ is the eigenvector of $\lambda\, \varphi = \boldsymbol{C}^\Phi\, \varphi$, where $\boldsymbol{C}^\Phi$ is the correlation matrix between the mapped data, namely $\boldsymbol{C}^\Phi = \frac{1}{n} \sum_{j=1}^n \Phi(\boldsymbol{x}_j)\, \Phi(\boldsymbol{x}_j)^\top$. In addition, each eigenvector φ lies in the span of the Φ-images, thus can be defined by some coefficients $\alpha_1, \alpha_2, \ldots, \alpha_n$ such that $\varphi = \sum_{i=1}^n \alpha_i\, \Phi(\boldsymbol{x}_i)$. It is easy to see that these coefficients are obtained by solving the eigen-problem: $\lambda\, \boldsymbol{\alpha} = \frac{1}{n} \boldsymbol{K}\, \boldsymbol{\alpha}$, where $\boldsymbol{\alpha} = [\alpha_1 \;\, \alpha_2 \;\, \cdots \;\, \alpha_n]^\top$ and $\boldsymbol{K}$ is a $n \times n$ matrix defined by $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}$, for $i, j = 1, 2, \ldots, n$. This allows to construct the most relevant subspace in the feature space $\mathcal{H}$, without the need to exhibit any of its elements.

In many pattern recognition applications, one needs to have an access on some elements of the RKHS. Consider the denoising problem using kernel PCA, where the relevant subspace is assumed to be a denoised subspace. Let $\tilde{\boldsymbol{x}}$ be a data corrupted by noise, and let $\Phi(\tilde{\boldsymbol{x}})$ be its image in the feature space. To provide a denoised version of the latter, it is projected onto the relevant subspace. Thus the resulting projection can be written as a linear expansion in terms of the n Φ-images, namely

$$\psi = \sum_{j=1}^n \gamma_j\, \Phi(\boldsymbol{x}_j), \tag{1}$$

where $\gamma_j = \sum_{k=1}^m \sum_{i=1}^n \alpha_{k,i}\, \alpha_{k,j}\, \kappa(\tilde{\boldsymbol{x}}, \boldsymbol{x}_i)$, with only m eigenvectors being retained [15]. This is illustrated in Figure 1.

3. PRE-IMAGE PROBLEM FOR PATTERN RECOGNITION: A GRADIENT APPROACH

Many pattern recognition techniques require, not only the feature (1) in the feature space, but also its counterpart in the input space, e.g., the signal space. Getting back, from the feature space to the input space, is the

Table 1: Gradient of the objective function (3) for most commonly used kernels, with respect to $\boldsymbol{x}$.

Kernel	Expression	$\nabla_{\boldsymbol{x}} J(\boldsymbol{x})$
Polynomial	$\kappa_p(\boldsymbol{x}_i, \boldsymbol{x}_j) = (c + \boldsymbol{x}_i \cdot \boldsymbol{x}_j)^p$	$\sum_{i=1}^{n} \gamma_i \, p \, \kappa_{p-1}(\boldsymbol{x}_i, \boldsymbol{x}) \, \boldsymbol{x}_i + p \, \kappa_{p-1}(\boldsymbol{x}, \boldsymbol{x}) \, \boldsymbol{x}$
Sigmoid	$\kappa_S(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tanh(c \, (\boldsymbol{x}_i \cdot \boldsymbol{x}_j) + \sigma)$	$\sum_{i=1}^{n} \gamma_i \, (1 - \kappa_S^2(\boldsymbol{x}_i, \boldsymbol{x})) \, c \, \boldsymbol{x}_i + c \, (1 - \kappa_S^2(\boldsymbol{x}, \boldsymbol{x})) \, \boldsymbol{x}$
Exponential	$\kappa_E(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(\frac{1}{\sigma}(\boldsymbol{x}_i \cdot \boldsymbol{x}_j))$	$\frac{1}{\sigma} \sum_{i=1}^{n} \gamma_i \, \kappa_E(\boldsymbol{x}_i, \boldsymbol{x}) \, \boldsymbol{x}_i + \frac{1}{\sigma} \, \kappa_E(\boldsymbol{x}, \boldsymbol{x}) \, \boldsymbol{x}$
Gaussian	$\kappa_G(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\frac{1}{2\sigma^2} \lVert \boldsymbol{x}_i - \boldsymbol{x}_j \rVert^2)$	$\frac{1}{\sigma^2} \sum_{i=1}^{n} \gamma_i \, \kappa_G(\boldsymbol{x}_i, \boldsymbol{x}) \, (\boldsymbol{x}_i - \boldsymbol{x})$

pre-image problem. This principle has shown its relevance in many signal processing techniques; see [13] for a recent review.

More precisely, we seek an element of the input space whose image in the feature space is defined by $\psi = \sum_{j=1}^{n} \gamma_j \Phi(\boldsymbol{x}_j)$. The element is the so-called *pre-image* of ψ. Nonetheless, such pre-image may not exist or it may not be unique. Thus, we seek an approximate pre-image $\boldsymbol{x}^*$ whose image $\Phi(\boldsymbol{x}^*)$ is as close as possible to ψ. This is the *pre-image problem*, defined by minimizing the distance between the corresponding features in the RKHS, namely

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \lVert \psi - \Phi(\boldsymbol{x}) \rVert_{\mathcal{H}}^2, \tag{2}$$

where $\lVert \cdot \rVert_{\mathcal{H}}$ denotes the norm in the RKHS.

In the general case, this optimization problem can be written as

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} J(\boldsymbol{x}),$$

where $J(\boldsymbol{x})$ is the objective function defined by

$$J(\boldsymbol{x}) = \sum_{i=1}^{n} \gamma_i \, \kappa(\boldsymbol{x}_i, \boldsymbol{x}) - \frac{1}{2} \kappa(\boldsymbol{x}, \boldsymbol{x}). \tag{3}$$

and the term $\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \gamma_i \, \gamma_j \, \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is removed since it is independent of $\boldsymbol{x}$. At the optimum, $\boldsymbol{x}^*$, the gradient of the aforementioned objective function goes to zero. Let $\nabla_{\boldsymbol{x}} J(\boldsymbol{x})$ denotes this gradient, namely

$$\nabla_{\boldsymbol{x}} J(\boldsymbol{x}) = \sum_{i=1}^{n} \gamma_i \frac{\partial \kappa(\boldsymbol{x}_i, \boldsymbol{x})}{\partial \boldsymbol{x}} - \frac{1}{2} \frac{\partial \kappa(\boldsymbol{x}, \boldsymbol{x})}{\partial \boldsymbol{x}}. \tag{4}$$

This is a general form for all kernels. This expression along with the objective function itself vary with the type of the kernel. Let us first consider the well-known Gaussian kernel. This so-called radial kernel, depending on the Euclidean distance $\lVert \boldsymbol{x}_i - \boldsymbol{x}_j \rVert$, is defined by

$$\kappa_G(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\frac{1}{2\sigma^2} \lVert \boldsymbol{x}_i - \boldsymbol{x}_j \rVert^2)$$

for any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^d$, where σ is a positive bandwidth parameter. The gradient, with respect to $\boldsymbol{x}$, of the resulting objective function is given by

$$\begin{aligned} \nabla_{\boldsymbol{x}} J(\boldsymbol{x}) &= \sum_{i=1}^{n} \gamma_i \frac{\partial \exp(-\frac{1}{2\sigma^2} \lVert \boldsymbol{x}_i - \boldsymbol{x} \rVert^2)}{\partial \boldsymbol{x}} \\ &= \frac{1}{\sigma^2} \sum_{i=1}^{n} \gamma_i \, \kappa_G(\boldsymbol{x}_i, \boldsymbol{x}) \, (\boldsymbol{x}_i - \boldsymbol{x}). \end{aligned}$$

Another used kernel is the polynomial, a direct generalization of the linear inner product. This projective kernel relies on the inner product $\boldsymbol{x}_i \cdot \boldsymbol{x}_j$. It is defined by $\kappa_p(\boldsymbol{x}_i, \boldsymbol{x}_j) = (c + \boldsymbol{x}_i \cdot \boldsymbol{x}_j)^p$, where $p \in \mathbb{N}^+$ and some positive c. In this case, we have

$$\nabla_{\boldsymbol{x}} J(\boldsymbol{x}) = \sum_{i=1}^{n} \gamma_i \, p \, \kappa_{p-1}(\boldsymbol{x}_i, \boldsymbol{x}) \, \boldsymbol{x}_i - p \, \kappa_{p-1}(\boldsymbol{x}, \boldsymbol{x}) \, \boldsymbol{x},$$

where $\kappa_{p-1}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (c + \boldsymbol{x}_i \cdot \boldsymbol{x}_j)^{p-1}$. Table 1 summarizes the gradient with respect to $\boldsymbol{x}$ of the most commonly used kernels.

We are now in a position to derive gradient-based algorithms to solve the pre-image problem, subject to the non-negativity constraint.

4. SOLVING THE PRE-IMAGE PROBLEM UNDER NON-NEGATIVITY CONSTRAINTS

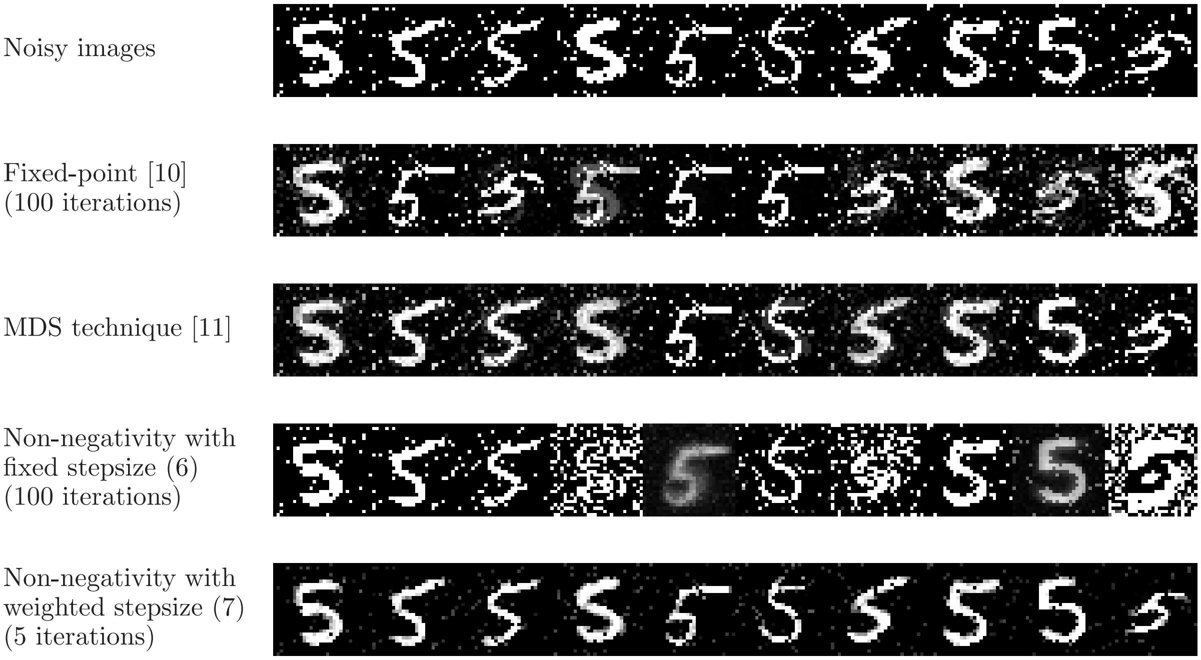
As aforementioned, some constraints are needed to ensure the physical interpretation of many signal and more precisely image processing techniques.

We consider the following constrained optimization problem:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} J(\boldsymbol{x})$$

$$\text{subject to } \boldsymbol{x} \geq 0$$

In this expression, the inequality defines the non-negativity of each component of the vector $\boldsymbol{x}$. Off-line optimization techniques are often computationally expensive and not efficient to solve this nonlinear optimization problem. Iterative techniques are essentially



Noisy images

Fixed-point [10] (100 iterations)

MDS technique [11]

Non-negativity with fixed stepsize (6) (100 iterations)

Non-negativity with weighted stepsize (7) (5 iterations)

Figure 2: A set of ten "5"-digit images corrupted by a salt-and-pepper noise of density 0.1 (first row), on which we applied the kernel PCA for data denoising. The pre-image results using the fixed-point iterative algorithm [10] are illustrated (second row), the MDS technique [11] (third row), the non-negative gradient pre-image (6) (fourth row) and the non-negative pre-image with the iterative schema (7) (last row).

based on a fixed-point approach, however only appropriate for convex objective function (see for instance [1]). However, our objective function (3) is non-convex, thus the fixed-point technique is not appropriate.

Next, we consider an iterative scheme, updating $\boldsymbol{x}(t+1)$ from $\boldsymbol{x}(t)$. Let $[\,\cdot\,]_\ell$ denotes the ℓ-th entry operator, and $x_\ell(t)$ the ℓ-th component of $\boldsymbol{x}(t)$, namely $[\boldsymbol{x}(t)]_\ell$. Component-wise, the gradient ascent scheme is defined by

$$x_\ell(t+1) = x_\ell(t) + \eta_\ell(t)\,[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell,$$

where $\eta_\ell(t)$ is a stepsize factor used to control convergence, and $[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell$ denotes the ℓ-th entry of the gradient of the objective function (3). A condition on $\eta_\ell(t)$ must be satisfied to ensure the non-negativity of $x_\ell(t+1)$ depending on the sign of the gradient. It is easy to see that, when the gradient is positive, there is no restrictions on the stepsize; when it is negative, the stepsize is upper-bounded by

$$\eta_\ell(t) \leq -\frac{x_\ell(t)}{[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell}.$$

While one can use a stepsize value for each direction in the gradient ascent algorithm, it is often interesting to have a single stepsize value at a given instance t. We shall define the stepsize $\eta(t)$ such that

$$\eta(t) \leq \min_\ell -\frac{x_\ell(t)}{[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell}. \tag{5}$$

This allows us to write the updating rule in matrix form, with

$$\boldsymbol{x}(t+1) = \boldsymbol{x}(t) + \eta(t)\,\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t)). \tag{6}$$

Next, we propose an approach to converge more rapidly toward null values. To this end, the stepsize $\eta_\ell(t)$ is weighted by the value of x_ℓ, which increases the speed to get towards zero. This leads to the expression

$$x_\ell(t+1) = x_\ell(t) + \eta_\ell(t)\,x_\ell(t)\,[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell.$$

The factor of convergence is now $\eta_\ell(t)\,x_\ell(t)$, which results in a new condition on $\eta_\ell(t)$ for the non-negativity of the solution. For this purpose, we write the above expression as

$$x_\ell(t+1) = x_\ell(t)\big(1 + \eta_\ell(t)[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell\big),$$

and obtain the non-negativity condition on $1 + \eta_\ell(t)\,[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell$. While no restriction is required when the gradient is positive, in the other case, when $[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell < 0$, the stepsize must be upper-bounded by $1/[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell$. In matrix form, the update rule can be written as

$$\boldsymbol{x}(t+1) = \boldsymbol{x}(t) + \eta(t)\,\mathrm{diag}[\boldsymbol{x}(t)]\,\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t)), \tag{7}$$

where $\mathrm{diag}[\boldsymbol{x}(t)]$ is a diagonal matrix whose entries are the elements of vector $\boldsymbol{x}(t)$. To impose the non-negativity, we consider a single stepsize value $\eta(t)$ with

$$\eta(t) \leq \min_\ell -\frac{1}{[\nabla_{\boldsymbol{x}} J(\boldsymbol{x}(t))]_\ell}. \tag{8}$$

Next, we study the performance of the gradient-based approach, with either the updating rule (7) (resp. (6)), and the non-negativity condition (8) (resp. (5)).

5. EXPERIMENTS

In this section, we illustrate the relevance of the proposed method in an image denoising task: real handwritten digits taken from the MNIST datasets[1]. We have chosen the digits "5". Each image is defined by grayscale 28 × 28 pixels, with values normalized between 0 and 1. Thus, each image can be written as a 784-dimensional vector. The images were corrupted by adding a Salt-and-Pepper noise, with 0.1 density. A set of 500 images was used to train the kernel PCA with 50 eigenvectors retained. Another set of 10 images, corrupted by the same noise settings, was used to demonstrate the relevance of this denoising technique.

We compared the proposed method to two state-of-the-art techniques: the MDS approach [11], and the fixed-point iterative method [10]. The latter is defined by the iterative expression

$$\boldsymbol{x}(t+1) = \frac{\sum_{i=1}^{n} \gamma_i \, \kappa_G(\boldsymbol{x}_i, \boldsymbol{x}(t)) \, \boldsymbol{x}_i}{\sum_{i=1}^{n} \gamma_i \, \kappa_G(\boldsymbol{x}_i, \boldsymbol{x}(t))}.$$

These methods have been successfully applied on many pattern recognition problems, mainly using the Gaussian kernel. For this reason, we consider the same kernel, the bandwidth was fixed for all algorithms ($\sigma = 8$). While the maximum number of iterations was set to 100, only 5 iterations were applied for the weighted stepsize algorithm.

We applied the aforementioned techniques along with the proposed algorithms. In Figure 2, the first row shows the noisy images of the digit "5". Results from the proposed method are given in the last two rows. They should be compared to the results obtained by the fixed-point iterative method and illustrated in the second row, and to the MDS technique illustrated in the third row. As we can see, the fixed point iterative algorithm failed in such applications. The MDS technique did not succeed in recognizing the patterns, as well as the simple gradient with a fixed stepsize. The proposed method, with the weighted stepsize, has shown to be relevant,, the patterns were recognized and all the handwritten digits identified.

6. CONCLUSION

In this paper, we studied the non-negativity of the pre-image in kernel machines. We showed that it is easy to impose the non-negativity using a simple gradient-based scheme. By considering a weighted stepsize in the iterative algorithm, the convergence speed was significantly improved, with remarkable self-regularization property. While the optimization problem is nonlinear and highly non-convex, the proposed technique gave extremely accurate results, with performance illustrated on handwritten digits taken from the MNIST datasets. In future work, we are interested in a general framework for nonlinear non-negative pattern recognition, including this work as well as our previous work [14]. In the latter, we considered the non-negativity in the coefficients of the linear model that defines the pre-image.

[1] The datasets are available from the address http://yann.lecun.com/exdb/mnist/.

REFERENCES

[1] H. Lantéri, M. Roche, O. Cuevas, and C. Aime, "A general method to devise maximum-likelihood signal restoration multiplicative algorithms with non-negativity constraints," *Signal Processing*, vol. 81, no. 5, pp. 945–974, May 2001.

[2] G. Thomas, "A positive optimal deconvolution procedure," *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, vol. 8, pp. 651 – 654, April 1983.

[3] R. Prost and R. Goutte, "Discrete constrained iterative deconvolution algorithms with optimized rate of convergence," *Signal Processing*, vol. 7, no. 3, pp. 209–230, Dec. 1984.

[4] G. Thomas and N. Souilah, "Utilisation des multiplicateurs de lagrange pour la restauration d'image avec contraintes," *Colloques sur le Traitement du Signal et des Images*, 1991.

[5] D.L. Snyder, T.J. Schulz, and J.A. O'Sullivan, "Deblurring subject to nonnegativity constraints," *IEEE Transactions on Signal Processing*, vol. 40, pp. 1143 – 1150, May 1992.

[6] D.D. Lee and H.S. Seung, "Algorithms for non-negative matrix factorization," *Advances in neural information processing systems*, pp. 556–562, 2001.

[7] J. Chen, C. Richard, P. Honeine, H. Lantéri, and C. Theys, "System identification under non-negativity constraints," in *Proc. EUSIPCO*, Aalborg, Denmark, 2010.

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

[9] V. Vapnik, *Statistical Learning Theory*, Wiley-Interscience, September 1998.

[10] S. Mika, B. Schölkopf, A. Smola, K. Müller, M. Scholz, and G. Rätsch, "Kernel PCA and de-noising in feature spaces," in *Proceedings of the 1998 conference on Advances in neural information processing systems II*, Cambridge, MA, USA, 1999, pp. 536–542, MIT Press.

[11] J. T. Kwok and I. W. Tsang, "The pre-image problem in kernel methods," *IEEE Trans. on Neural Networks*, vol. 15, no. 6, pp. 1517–1525, November 2004.

[12] P. Honeine and C. Richard, "Solving the pre-image problem in kernel machines: a direct method," in *Proc. IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, Grenoble, France, September 2009, (best paper award).

[13] P. Honeine and C. Richard, "Pre-image problem in kernel-based machine learning," in *IEEE Signal Processing Magazine*, March 2011, vol. 28 (2), pp. 77–88.

[14] M. Kallas, P. Honeine, C. Richard, H. Amoud, and C. Francis, "Nonlinear feature extraction using kernel principal component analysis with non-negative pre-image," in *Proc. 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Buenos Aires, Argentina, 31 Aug. - 4 Sept. 2010.

[15] B. Schölkopf, S. Mika, A. Smola, G. Rätsch, and K.-R. Müller, "Kernel pca pattern reconstruction via approximate pre-images," in *Proc. 8th International Conference on Artificial Neural Networks*, 1998, pp. 147–152.