Journal of Mathematical Imaging and Vision manuscript No.
(will be inserted by the editor)

Fast Kernel Generalized Discriminative Common Vectors for Feature Extraction

Katerine Diaz-Chito · Jesús Martínez del Rincón · Aura Hernández-Sabaté · Marçal Rusiñol · Francesc J. Ferri

Received: date / Accepted: date

Abstract This paper presents a supervised subspace learning method called Kernel Generalized Discriminative Common Vectors (KGDCV), as a novel extension of the known Discriminative Common Vectors method with Kernels. Our method combines the advantages of kernel methods to model complex data and solve nonlinear problems with moderate computational complexity, with the better generalization properties of generalized approaches for large dimensional data. These attractive combination makes KGDCV specially suited for feature extraction and classification in computer vision, image processing and pattern recognition applications. Two different approaches to this generalization are proposed, a first one based on the kernel trick (KT) and a second one based on the nonlinear projection trick (NPT) for even higher efficiency. Both methodologies have been validated on four different image datasets containing faces, objects and handwritten digits, and compared against well known non-linear state-of-art methods. Results show better discriminant properties than other generalized approaches both linear or kernel. In addition, the KGDCV-NPT approach presents a considerable computational gain, without compromising the accuracy of the model.

Katerine Diaz-Chito, Aura Hernández-Sabaté, Marçal Rusiñol
Centre de Visió per Computador, Universitat Autònoma de Barcelona, Spain
E-mail: {kdiaz, aura, marcal}@cvc.uab.es

Jesús Martínez del Rincón
Centre for Secure Information Technologies, Queen's University Belfast, UK
E-mail: j.martinez-del-rincon@qub.ac.uk

Francesc J. Ferri
Departament d'Informàtica, Universitat de Valencia, Spain
E-mail: francesc.ferri@uv.es

Keywords Nonlinear feature extraction · Kernel Discriminative Common Vectors · Kernel Trick · Nonlinear Projection Trick · Computational Efficiency.

1 Introduction

Statistical methods based on subspaces are extensively applied in computer vision and machine learning [2,6, 7,8]. In particular, their role as dimensionality reduction and automatic feature extraction tools has been crucial to mitigate the curse of dimensionality inherent to image classification. Thus, subspace methods are often used as preprocessing or feature selection stages in order to facilitate learning by the classifier. In this image context, the ratio between the dimensionality of the input space and the training set size is usually very large. This unbalance ratio poses serious difficulties in the application of classifiers and machine learning algorithms and affects their generalization ability and efficiency. This situation is usually called the *Small Sample Size* (SSS) problem [11]. While it can be mitigated by acquiring larger datasets to balance the ratio, this is not always possible.

As a consequence, there has been a need to develop methods able to work under these constraints. Cevikalp et al. [6] proposed a supervised method called Discriminant Common Vector (DCV) to specifically address the SSS, based on a modified Fisher's linear discriminant criterion [2]. This approach divides the feature space into the range and the null subspaces, being the later important for extracting useful discriminative features for recognition. However, it can only be applied when the number of samples is smaller than the dimensionality of the data. An extension of the DCV method, the Generalized Discriminant Common Vec-

tor (GDCV), also called Rough Common Vector (RCV) was presented by Akihiko et al. [25], where the bases of the method are reinterpreted and some assumptions are relaxed. The GDCV method exhibits better generalization properties in a wider range of applications, and it can be applied for both small and large datasets regarding the dimension of the input space. As main limitation, these methodologies do not perform well when samples belonging to different classes are not separable using linear transformations.

Traditionally, the Kernel Trick (KT) has been widely used to extend linear methods to the nonlinear case [23, 21]. Kernel methods have aroused great interest in the last decade since they are universal nonlinear approximators and facilitate solving complex problems where the samples are not linearly separable as is the case of many machine learning and pattern recognition application. Kernel methods use nonlinear mapping to project samples from the original space to a feature space where the samples are expected to be easily separable, as depicted by the example in Figure 1. Specifically, the KT operates by calculating an implicit projection to a space of greater or even infinite dimension, where the linear discriminant can be effectively applied to separate the originally non-linear separable classes. Moreover, the KT makes the kernel methods computationally efficient in comparison with other nonlinear techniques, since the nonlinear mapping function and the mapped samples are not used explicitly. They, however, require selecting an appropriate kernel function, which must be carefully chosen in every application to avoid numerical instabilities and overfitting problems, as well as problems associated to handling large datasets. For details of how to select and tune kernel functions to particular applications see [24].

By applying the KT approach, a variety of subspace based kernel methods have been proposed, including Kernel Principal Component Analysis (KPCA) and Kernel Discriminant Analysis (KDA) [28]. A Kernel Independent Component Analysis (KICA) [17] by using the KT and the InfoMax algorithm has also been proposed for enhancing classification, but its application is limited to classes statistically independent. KDA-based approaches are better suited for supervised classification applications since a similar supervision process is performed during the dimensionality reduction, but they require solving an expensive optimisation problem [3]. Efficient KDA approaches have been proposed as a solution such as the Kernel Discriminant Analysis via QR decomposition (KDAQR) [27], based on the QR decomposition to replace the costly eigendecomposition of the kernel matrix, and the Kernel Discriminant Analysis by using Spectral Regression (KDASR) [3] which

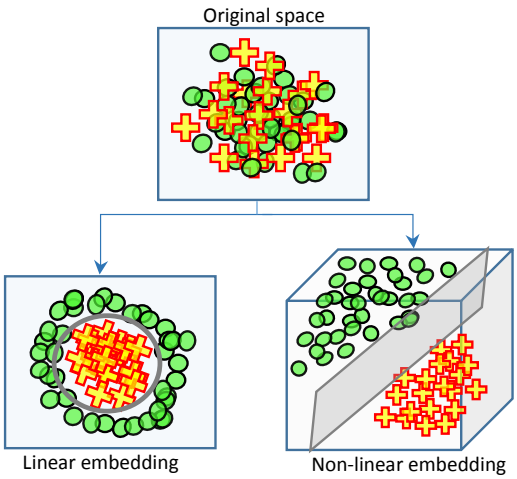


Fig. 1: Example of non-linearly separable data and its mapping into linearly separable space through a non-linear kernel.

combines spectral graph analysis and regularised regression. Finally, a Discriminative Common Vector with Kernel (KDCV) was originally proposed by Cevikalp et al. [5,4] and extended for efficient implementation in [29,27], at the expense of an slight numerical instability. However, all these techniques inherited the corresponding DCV restrictions for small datasets regarding the dimensionality of the samples.

In this paper we aim to combine the advantages of KDCV for non-linear spaces and GDCV for better generalisation properties without restrictions on the training set size. Thus, the novel Kernel Generalized Discriminative Common Vectors (KGDCV) is introduced by extending the GDCV with kernels. This non linear extension is first achieved by applying the kernel trick. A second alternative is also proposed based on the Nonlinear Projection Trick (NPT) [16]. NPT explicitly maps the input data into a reduced dimensional kernel Hilbert space, using the eigenvalue decomposition of the kernel matrix. Our approach is evaluated using a range of different image classification problems and datasets, including facial images, objects and handwritten digits, that allow us to analyse its behaviour against different training set sizes. A comparison study is performed in these scenarios against the conventional linear/non-linear DCV-based methods and the well-known KICA [17], KDAQR [27] and KDASR [3] methods.

The remainder of the paper is structured as follows. Section 2 briefly introduces linear and nonlinear DCV as background information. Section 3 presents the novel KGDCV using both KT and NPT approaches, as main contributions of this paper. Section 4 describes the empirical validation and comparison with the state of the

art, presents the results and analyse the proposed approach. Finally, Section 5 summarizes the main conclusions. A list of acronyms is presented in Table 1 of the appendix A.

2 Background

Let X be the training set composed of $m = \sum_{j=1}^{c} m_j$ samples belonging to c classes, where every class j has m_j samples. Let x_j^i be a d-dimensional column vector which denotes the i^{th} sample from the j^{th} class.

2.1 Linear DCV

In order to obtain the optimal projection W of the samples X to the new subspace, the bases of such subspace U should be first calculated. These bases are obtained by solving the eigenproblem of the within-scatter matrix,

$$S_w^X = \sum_{j=1}^{c} \sum_{i=1}^{m_j} (x_j^i - \overline{x}_j)(x_j^i - \overline{x}_j)^T = X_c {X_c}^T \tag{1}$$

where $\overline{x}_j$ is the average of the samples in the jth class, and the centered data matrix, X_c, consists of column vectors $(x_j^i - \overline{x}_j)$ for all $j = 1 \dots c$ and $i = 1 \dots m_j$.

The eigendecomposition or eigen-value/vector decomposition (EVD) of S_w^X can be written in general as:

$$\begin{aligned} EVD(S_w^X) : X_c {X_c}^T &= U \Lambda U^T \\ &= [U_r \;\; U_o] \begin{bmatrix} \Lambda_r & \\ & 0 \end{bmatrix} \begin{bmatrix} {U_r}^T \\ {U_o}^T \end{bmatrix} \end{aligned} \tag{2}$$

where $U = [u_1 \dots u_d]$ is a column matrix formed by the eigenvectors associated to the eigenvalues, $\lambda_1 \geq \dots \geq \lambda_d$, contained in the diagonal matrix Λ. r is the range of matrix S_w^X, that is, $\lambda_i = 0$ for all $i > r$.

By decomposing U into two matrices, U_r containing the eigenvectors associated to the non-zero eigenvalues, and U_o containing the eigenvectors associated to the zero-value eigenvalues, DCV is able to separate the feature space into two complementary subspaces, the range space, $\mathcal{R}(S_w^X)$ with bases U_r, and the null space, $\mathcal{N}(S_w^X)$ with bases U_o, respectively.

The DCV approach is an effective method for solving the SSS problem. If S_w^X is singular, all samples x_j^i belonging to class j can be mapped to a common vector $x_{cv}^j = \overline{x}_j - U_r U_r^T \overline{x}_j$ in the null space. This extract the common properties of classes in the training set by eliminating the differences of the samples in each class, i.e. the features that are in the direction of the eigenvectors corresponding to the nonzero eigenvalues of the within-class scatter matrix.

For classification, the centered version X_c^{com} of $X^{com} = [x_{cv}^1 \dots x_{cv}^c]$, with regard to the mean $\overline{x}_{com} = (1/c) \sum_{j=1}^{c} x_{cv}^j$, is calculated to compute the final projection matrix, $W = orth(X_c^{com}) \in \mathbb{R}^{d \times (c-1)}$, and obtain the discriminative common vectors as $W^T \overline{x}_j$.

2.2 Linear GDCV

DCV can not be applied when $d < (m-c)$, i.e. the number of samples is bigger than their dimensionality. This case would lead to a non-singular within-class scatter matrix, where the null space does not exist. Even if the within-class scatter matrix is singular, the recognition rate of the DCV may not be good if the dimensionality of the null space is small. This SSS singularity problem [14] is avoided by extending the null space to include not only null directions or basis vectors, i.e. $\lambda_i = 0$, but also a set of almost null directions, $\lambda_i \approx 0$. This extension of the null space also implies the corresponding restriction of the range space. The projection basis U_α of the new restricted range space will be the basis of the learned subspace.

The scattering added to the null space is measured by the trace $tr(\cdot)$ as $tr(U_\alpha^T S_w^X U_\alpha)$. This quantity is at most $tr(S_w^X)$ when no directions are removed, $U_\alpha = U_r$, and decreases as more and more important directions disappear from U_r. Consequently, the scattering preserved after a projection, U_α, is written as follows

$$\alpha = 1 - \frac{tr(U_\alpha^T S_w^X U_\alpha)}{tr(S_w^X)} \tag{3}$$

The parameter α takes values within the interval $[0, 1]$. When $\alpha = 0$, then $U_\alpha = U_r$. For individual values of $0 < \alpha < 1$, different projections are obtained with dissimilar levels of preserved variability. Figure 2 presents the main subspaces involved in the DCV and GDCV method.

Once U_α is calculated the generalized common vectors are defined as $x_{gcv}^j = \overline{x}_j - U_\alpha U_\alpha^T \overline{x}_j$. Then the centered generalized common vectors $X_c^{com} = [x_{gcv}^1 - \overline{x}_{com} \;\; \dots \;\; x_{gcv}^c - \overline{x}_{com}]$, with regard to the mean $\overline{x}_{com} = (1/c) \sum_{j=1}^{c} x_{gcv}^j$, is calculated to compute the final projection matrix as in the DCV method.

To test a new sample, x_{test}, we need to project it on W^T ($W^T x_{test}$) and then the label is allocated according to the minimum distance between the projected sample and the generalized discriminative common vectors.

Regarding its computational complexity, GDCV has an asymptotic cost dominated by $O(d^2 m + d^3)$, when $d \leq m$. In the SSS case, $d > m$, the computational complexity is $O(dm^2 + m^3 + dmr)$.

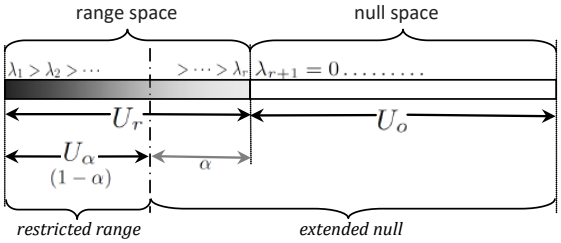


Fig. 2: Main subspaces involved in the DCV and GDCV methods. U_r and U_o span the range and null space of S_w^X linked to the eigenvalues $\lambda_1 > \ldots > \lambda_r$ and $\lambda_i = 0, \quad i \geq (r+1)$, respectively. U_α spans the restricted range of S_w^X according to α.

2.3 KDCV by using Kernel Trick

The KDCV method [4,5] uses nonlinear mapping to map samples from the original input space $\mathbb{R}^d$ to a feature space $\mathbb{R}^f$ of greater dimension where the DCV method is applied and the samples are expected to be linearly separable.

Given the nonlinear function ϕ, the re-projected version of the training set X is defined as:

$$\Phi(X) = [\phi(x_1^1) \ \ldots \phi(x_1^{m_1}) \ \phi(x_2^1) \ \ldots \phi(x_j^{m_j}) \ldots \phi(x_c^{m_c})] \tag{4}$$

In this new space, the between-class, the within-class and the total scatter matrices are defined as S_B^ϕ, S_W^ϕ and S_T^ϕ, respectively.

$$\begin{aligned} S_B^\Phi &= \sum_{j=1}^{c} m_j(\overline{x}_j^\phi - \overline{x}^\phi)(\overline{x}_j^\phi - \overline{x}^\phi)^T \\ &= (\Phi H - \Phi L)(\Phi H - \Phi L)^T \end{aligned} \tag{5}$$

$$\begin{aligned} S_W^\Phi &= \sum_{j=1}^{c}\sum_{i=1}^{m_j}(\phi(x_j^i) - \overline{x}_j^\phi)(\phi(x_j^i) - \overline{x}_j^\phi)^T \\ &= (\Phi - \Phi G)(\Phi - \Phi G)^T \end{aligned} \tag{6}$$

$$\begin{aligned} S_T^\Phi &= \sum_{j=1}^{c}\sum_{i=1}^{m_j}(\phi(x_j^i) - \overline{x}^\phi)(\phi(x_j^i) - \overline{x}^\phi)^T \\ &= (\Phi - \Phi 1_m)(\Phi - \Phi 1_m)^T = S_W^\Phi + S_B^\Phi \end{aligned} \tag{7}$$

where $\overline{x}_j^\phi$ are the re-projected averages for each j^{th} class and $\overline{x}^\phi$ is the global average of all re-projected samples in $\mathbb{R}^f$. $G = diag[G_1, \ldots, G_c] \in \mathbb{R}^{(m \times m)}$ and $H = diag[\mu_1, \ldots, \mu_c] \in \mathbb{R}^{(m \times c)}$ are diagonal matrices, where each $G_j \in \mathbb{R}^{(m_j \times m_j)}$ is a matrix with all its elements equal to $1/m_j$, and each $\mu_j \in \mathbb{R}^{(m_j \times 1)}$ is a vector with all its elements equal to $1/\sqrt{m_j}$. $L = [l_1, \ldots, l_c] \in \mathbb{R}^{(m \times c)}$ is a matrix where each $l_j \in \mathbb{R}^{(m \times 1)}$ is a vector with all its elements equal to $\sqrt{m_j}/m$, and $1_m \in \mathbb{R}^{(m \times m)}$ is a matrix with all its elements equal to $1/m$.

KDCV uses the intersection between the null subspace of S_W^ϕ and the range subspace of S_T^ϕ to represent classes [4,5]. Therefore, the common vectors are calculated from a first re-mapping on the range space of S_T^ϕ (which is nothing more than the application of the principal component analysis with kernel o KPCA(S_T^ϕ): $V\Delta V^T$ [22]) followed by a re-mapping onto the null subspace of S_W^ϕ, obtaining the nonlinear discriminant common vectors representing each class. The mathematics properties of the DCV method are transmitted to the KDCV method, only differing in the mapping of the samples, as follows:

$$\begin{aligned} \tilde{S}_W^\Phi &= ((\Phi - \Phi 1_m)V\Delta^{-1/2})^T S_W^\Phi (\Phi - \Phi 1_m)V\Delta^{-1/2}, \\ &= \Delta^{-1/2}V^T\tilde{K}_W\tilde{K}_W^T V\Delta^{-1/2}. \end{aligned} \tag{8}$$

$$\begin{aligned} \tilde{S}_B^\Phi &= ((\Phi - \Phi 1_m)V\Delta^{-1/2})^T S_B^\Phi (\Phi - \Phi 1_m)V\Delta^{-1/2}, \\ &= \Delta^{-1/2}V^T\tilde{K}_B\tilde{K}_B^T V\Delta^{-1/2}. \end{aligned} \tag{9}$$

$$\begin{aligned} \tilde{S}_T^\Phi &= ((\Phi - \Phi 1_m)V\Delta^{-1/2})^T S_T^\Phi (\Phi - \Phi 1_m)V\Delta^{-1/2}, \\ &= \Delta^{-1/2}V^T V\Delta V^T V\Delta V^T V\Delta^{-1/2} = \Delta, \end{aligned} \tag{10}$$

with $\tilde{K}_W = K - KG - 1_m K + 1_m KG = (K - 1_m K)(I - G)$ and $\tilde{K}_B = KH - KL - 1_m KH + 1_m KL = (K - 1_m K)(H - L)$. K is the kernel matrix of the mapped data $K = \Phi^T\Phi$. $\tilde{K} = K - 1_m K - K1_m + 1_m K1_m$ is the centered training kernel, and $(\Phi - \Phi 1_m)V\Delta^{-1/2}$ is the transformation matrix that maps the training set into $R(S_T^\Phi)$.

An EVD of $\tilde{S}_W^\Phi$ is then performed to obtain the null subspace base $\tilde{U}_o$, such that $EVD(\tilde{S}_W^\Phi) : \tilde{U}\tilde{\Lambda}\tilde{U}^T = [\tilde{U}_r \ \tilde{U}_o] \ diag(\tilde{\Lambda}_r, \tilde{\Lambda}_o) \ [\tilde{U}_r \ \tilde{U}_o]^T$. $\tilde{U}_o$ are the r_o normalized eigenvectors associated to the null eigenvalues in $\tilde{\Lambda}$, such that

$$\tilde{U}_o^T \ \tilde{S}_B^\Phi \ \tilde{U}_o = \tilde{U}_o^T \ \tilde{S}_T^\Phi \ \tilde{U}_o. \tag{11}$$

The re-mapping matrix W is calculated as: $W = (\Phi - \Phi 1_M)V\Delta^{-1/2}\tilde{U}_o$, and the nonlinear discriminative common vectors are obtained by: $x_{kdcv}^j = W^T\overline{x}_j^\phi = (V\Delta^{-1/2}\tilde{U}_o)^T\tilde{K}$.

Regarding its computational complexity, the asymptotic cost of the KDCV-KT is dominated by $O(9m^3)$.

3 Kernel Generalized Discriminative Common Vectors

In this section we present the main two contributions of this paper, which are the extension of the GDCV for non linear cases using the Kernel Trick first, and by means of the Nonlinear Projection Trick later.

3.1 KGDCV by applying the Kernel Trick

Equation (11) is not true for $0 < \alpha < 1$ values, since $[\tilde{U}_o\ \tilde{U}_{r'}]^T\ \tilde{S}_B^{\Phi}\ [\tilde{U}_o\ \tilde{U}_{r'}] \neq [\tilde{U}_o\ \tilde{U}_{r'}]^T\ \tilde{S}_T^{\Phi}\ [\tilde{U}_o\ \tilde{U}_{r'}]$. For convenience, let us use $\tilde{U}_{1-\alpha} = [\tilde{U}_o\ \tilde{U}_{r'}]$ to denote the extended null space, similarly to $\tilde{U}_\alpha$ was used to denote the restricted range space. $\tilde{U}_{1-\alpha}$ spans the null space with the normalized eigenvectors $\tilde{U}_o$ of $\tilde{S}_W^{\Phi}$ associated to the null eigenvalues plus the normalized eigenvectors $\tilde{U}_{r'}$ associated to the smallest r' non-zero eigenvalues. In this case the dimension of W will not be limited to $(c-1)$ and it will grow rapidly with α.

To avoid this rapid grow and limit the final dimension to $(c-1)$, the $\tilde{U}_{1-\alpha}^T\ \tilde{S}_B^{\phi}\ \tilde{U}_{1-\alpha}$ matrix is eigendecomposed in $Y\tilde{\Delta}Y^T$, as in [4]. This is equivalent to consider only the average vectors of each class $\overline{x}_j^{\phi}$ in the high-dimensional space defined by the kernel ϕ. In this way, the final re-mapping matrix is defined, similarly to KDCV, as:

$$W = (\Phi - \Phi 1_M)V\Delta^{-1/2}\tilde{U}_{1-\alpha}Y. \tag{12}$$

When new samples need to be re-projected in the subspace, as the testing samples x_{test} in a classification pipeline, the kernel matrix K^{test} is first calculated with entries $k(x_j^i, x_{test}) =< \phi(x_j^i), \phi(x_{test}) >$. Then, the test sample can be re-projected as:

$$x_{test}^{\phi} = (V\Delta^{-1/2}\tilde{U}_{1-\alpha}Y)^T(K^{test} - K1'_m - 1_m K^{test} + 1_m K1'_m) \tag{13}$$

where $1'_m = (1/m)_{(m\times p)}$.

Conventionally, if a nearest neighbor classifier is applied, the test label is allocated from the minimum distance between the re-projected sample x_{test}^{ϕ} and the nonlinear generalized discriminative common vectors x_{kgdcv}^j:

$$label_{test} = \arg\min_{j\in[1,c]} \|x_{test}^{\phi} - x_{kgdcv}^j\| \tag{14}$$

Alternatively, all training samples x_j^i can be re-projected to take into account the variability of each class in the new space, and then the classifier is used.

The KGDCV algorithm is presented in Algorithm 1.

The asymptotic cost of the KGDCV-KT is dominated by $O(9m^3)$, like in the KDCV method.

3.2 KGDCV by applying the Nonlinear Projection Trick

While KGDCV-KT allows both addressing non-linearly separable spaces and dealing with the SSS singularity, it also implies an increase in the computational complexity regarding the linear case. As alternative, we propose a second KGDCV method that applies the Nonlinear Projection Trick (NPT) [16] to compute the same feature space in a more efficient manner. Thus, our KGDCV explicitly maps the input space into the reduced kernel feature space. This is achieved by the eigenvalue decomposition of the kernel matrix that allows deriving an exact coordinates of the mapped input data.

N. Kwak [16] demonstrated that applying a machine learning algorithm to the re-projected data on a kernel Hilbert space $\Phi(X)$, whose coordinates Xo are obtained using the NPT, is equivalent to apply a kernel version of the machine learning algorithm to the original data in the input space. Therefore, applying the linear method to Xo is equivalent to apply the kernel method to X, $GDCV(Xo) \equiv KGDCV(X)$.

Let Γ be an r-dimensional subspace of the feature space formed by the mapped training samples $\Phi(X)$. The columns of $\beta = \Phi(X)V\Delta$ constitute an orthonormal base of Γ, where V and Δ are obtained by the eigendecomposition of $K(X) =< \Phi(X),\ \Phi(X) >= V\Delta V^T$, such that the exact coordinate Xo of $\Phi(X)$ pro-

Algorithm 1 *KGDCV Algorithm by using Kernel Trick*

Parameter: $\alpha,\ 0 \leq \alpha < 1$
Input: $X \in \mathbb{R}^{d\times m}$, $m = \sum_{j=1}^{c} m_j$
Output: $W \in \mathbb{R}^{d\times(c-1)}$

TRAINING

1. *Compute the kernel matrix* $K = \Phi^T\Phi$ *(usually a radial kernel) with entries* $k(x_i, x_j) =< \phi(x_i),\ \phi(x_j) >= exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$.
2. *Center the training kernel* $\tilde{K} = K - 1_m K - K1_m + 1_m K1_m$.
3. *Calculate the normalized eigenvectors associated with the nonzero eigenvalues of* $\tilde{K}$*, such that EVD(*$\tilde{K}$*):* $V\Delta V^T \in \mathbb{R}^{(m\times m)}$.
4. *Calculated* $\tilde{S}_W^{\Phi} = \Delta^{-1/2}V^T\tilde{K}_W\tilde{K}_W^T V\Delta^{-1/2}$.
5. *Calculated* $\tilde{U}_{1-\alpha}$ *from EVD(*$\tilde{S}_W^{\Phi}$*).*
6. *Calculate* Y *from EVD(*$\tilde{U}_{1-\alpha}^T\tilde{S}_B^{\phi}\tilde{U}_{1-\alpha}$*):* $Y\tilde{\Delta}Y^T$.
7. *Calculate the final re-mapping matrix as* $W = (\Phi - \Phi 1_m)V\Delta^{-1/2}\tilde{U}_{1-\alpha}Y$.
8. *Obtain the nonlinear generalized discriminative common vectors as* $x_{kgdcv}^j = (V\Delta^{-1/2}\tilde{U}_{1-\alpha}Y)^T\tilde{K}_j$.

TESTING

Given a new testing sample x_{test}*,*

9. *Compute the kernel matrix* $K^{test}(x_j^i, x_{test}) = exp\left(-\frac{\|x_j^i - x_{test}\|^2}{2\sigma^2}\right)$.
10. *Center the testing kernel* $\tilde{K}^{test} = K^{test} - K1'_m - 1_m K^{test} + 1_m K1'_m$.
11. *Map the testing sample as* $x_{test}^{\phi} = (V\Delta^{-1/2}\tilde{U}_{1-\alpha}Y)^T\tilde{K}^{test}$
12. *Predict the classification label, normally as:*

$$label_{test} = \arg\min_{j\in[1,c]} \|x_{test}^{\phi} - x_{kgdcv}^j\|.$$

jected onto Γ is obtained by the inner product of the β and $\Phi(X)$ as:

$$Xo =< \beta,\ \Phi(X) >= \Delta^{-1/2} V^T K(X) \tag{15}$$

The full KGDCV-NPT algorithm is described in Algorithm 2.

Algorithm 2 *KGDCV Algorithm by using Nonlinear Projection Trick*

Parameter: $\alpha,\ 0 \leq \alpha < 1$
Input: $X \in \mathbb{R}^{d \times m}$, $m = \sum_{j=1}^{c} m_j$
Output: $W \in \mathbb{R}^{d \times (c-1)}$, $\Delta \in \mathbb{R}^{r \times r}$, $V \in \mathbb{R}^{m \times r}$

TRAINING

1. *Compute the kernel matrix* $K = \Phi^T \Phi$ *(usually a radial kernel) with entries* $k(x_i, x_j) =< \phi(x_i),\ \phi(x_j) >= exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$.
2. *Center the training kernel* $\tilde{K} = K - 1_m K - K 1_m + 1_m K 1_m$.
3. *Calculate the normalized eigenvectors associated with the nonzero eigenvalues of* $\tilde{K}$*, such that EVD(*$\tilde{K}$*):* $V \Delta V^T \in \mathbb{R}^{(m \times m)}$.
4. *Calculate the coordinates* $Xo = \Delta^{-1/2} V^T \tilde{K}$.
5. *Calculate GDCV(Xo):* (W, x^j_{gcv}).
6. *Obtain the nonlinear generalized discriminative common vectors as* $x^j_{kgdcv} = W^T x^j_{gcv}$.

TESTING

Given a new testing sample x_{test},

7. *Compute the kernel matrix* $K^{test}(x^i_j, x_{test}) = exp\left(-\frac{\|x^i_j - x_{test}\|^2}{2\sigma^2}\right)$.
8. *Center the testing kernel* $\tilde{K}^{test} = K^{test} - K 1'_m - 1_m K^{test} + 1_m K 1'_m$.
9. *Calculate the coordinate* $Xo^{test} = \Delta^{-1/2} V^T \tilde{K}^{test}$
10. *Compute the generalized discriminant features as* $x^{\phi}_{test} = W^T Xo^{test}$.
11. *Predict the classification label, normally as:*

$$label_{test} = \arg \min_{j \in [1,c]} \|x^{\phi}_{test} - x^j_{kgdcv}\|.$$

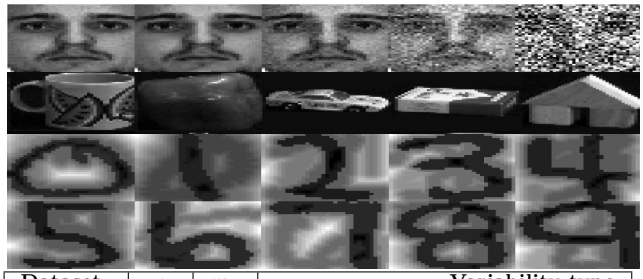
The asymptotic cost of the KGDCV-NPT is dominated by $O(5m^3)$, from the eigendecomposition of $\tilde{K}$ and the $GDCV(Xo)$.

4 Experiments and Results

In this section we present the experimental results carried on to validate our proposed approaches.

4.1 Datasets and Experimental setup

In our experimental setup, a simple 1-Nearest Neighbors classifier is employed as classifier, using the Euclidean distance between the trained nonlinear generalized discriminative common vectors and the projected test samples, as described in eq. 14. The simplicity of the classifier is justified for our aim to demonstrate the accuracy and approximation of our method to obtain a re-mapping into another space where the relevant information is easily separable into the different classes. This choice is also supported by the literature [16, 25, 4, 5] as a common practice. To validate the advantages of our KGDCV approaches, we have selected four publicly available image classification datasets containing faces, objects and handwritten digits for training and testing. The table in Figure 3 shows the main characteristics of the datasets.



Dataset	c	m_j	Variability type
ALL-NG	95	50	Faces - pose & light & noise
COIL-100	100	72	Objects - pose
NIST	10	100	handwritten digits - different writers
USPS	10	100	handwritten digits - different writers

Fig. 3: Datasets used in our evaluation along with their corresponding details. c is the number of classes. m_j is the number of samples per class.

ALL-NG is composed of a random selection of 10 face samples per class from the facial databases AR Face [18], ORL [20], Yale [12] and UMIST [26], totaling 95 classes. Images were resized to 40×40 pixels and they include changes of expression, lighting and pose. Gaussian noise with 4 different variance levels -0.02, 0.04, 0.06 and 0.08- was added to the 10 standardized original images of each subject, generating a total of 50 samples per class.

COIL-100 [19] is a database comprising 100 different classes of objects. Each class contains 72 grayscale images of the same object from poses 0 to 355 with a pose interval of 5 degrees. Images have a resolution of 128×128.

NIST is a database of gray-scale handwritten digits from 0 to 9. Image resolution is 32×32 This is derived from the NIST32 database available in prtools [13]. For each class, 100 samples were randomly selected from the total like in [10]. The original binary images were converted to gray levels using the distance transformation [1, 15].

USPS dataset [9] is a numerical dataset collected by scanning handwritten digits from envelopes by the U.S.

Postal Service. The original scanned digits are binary and have different sizes and orientations. The images were converted to gray-scale, as in NIST, and resized to 16×16.

As kernel function, a Gaussian radial kernel function is used in all experiments as in [27,5,3,16,29]. In order to prevent any overfitting to our particular method on the experimentation, the kernel's proximity parameter was varied in the range of 10 to 200 and optimised for the KDCV method before applying our approach. The empirical values obtained were $\sigma = 65$ for ALL-NG, $\sigma = 170$ for COIL-100, $\sigma = 25$ to NIST and $\sigma = 20$ to USPS.

To validate the discriminative properties and compare the computational efficiency of the methods, three experiments are considered:

1. **KGDCV performance analysis over both the training set size and the α value**: the accuracy rate for KGDCV is computed as a function of both parameters, the training set size and the α value.
2. **Comparative analysis in the SSS vicinity over α**: Our KGDCV-KT and KGDCV-NPT are compared against the linear methods DCV and GDCV and non-linear KDCV, both in terms of CPU time and classification accuracy. A small fixed training set is chosen so that the generalization ability of the method can be evaluated in the vicinity of the SSS problem, SSS singularity and low dimensional null spaces. Specifically, $m_j = 13$ per class is chosen for all datasets. Performance is analysed as a function of the variance added to the final subspace α, decreasing from 0 to 0.5 in steps of 0.05.
3. **Overall comparative analysis over the training size**: Our KGDCV-KT and KGDCV-NPT are compared against the linear method GDCV and state-of-art non-linear methods KDCV, KICA [17], KDAQR [27] and KDASR [3], both in terms of CPU time and classification accuracy. To validate the performance in all possible cases, the training set was varied from 3 samples per class up to the maximum. α value was chosen empirically from the previous experiment as a good compromise between time and accuracy ($\alpha_{ALL-NG} = 0.1$, $\alpha_{COIL-100} = 0.05$, $\alpha_{NIST} = 0.15$, and $\alpha_{USPS} = 0.05$).

Training set is composed by the 70% of the samples of each class, and the remaining 30% is used as test set. In the last two scenarios, cross validation is applied as evaluation protocol to avoid bias to a particular training/testing split, where the experiment is run 10 times with different random training/testing sample choices. Graphs show the average result over the iterations as well as dispersion bars. All algorithms have been run on a computer with an Intel(R) Core(TM) i7-4790 CPU @ 3.60GHz, 3601 Mhz, and 32-GB RAM.

4.2 Results

In the following we present the results for the three experiments.

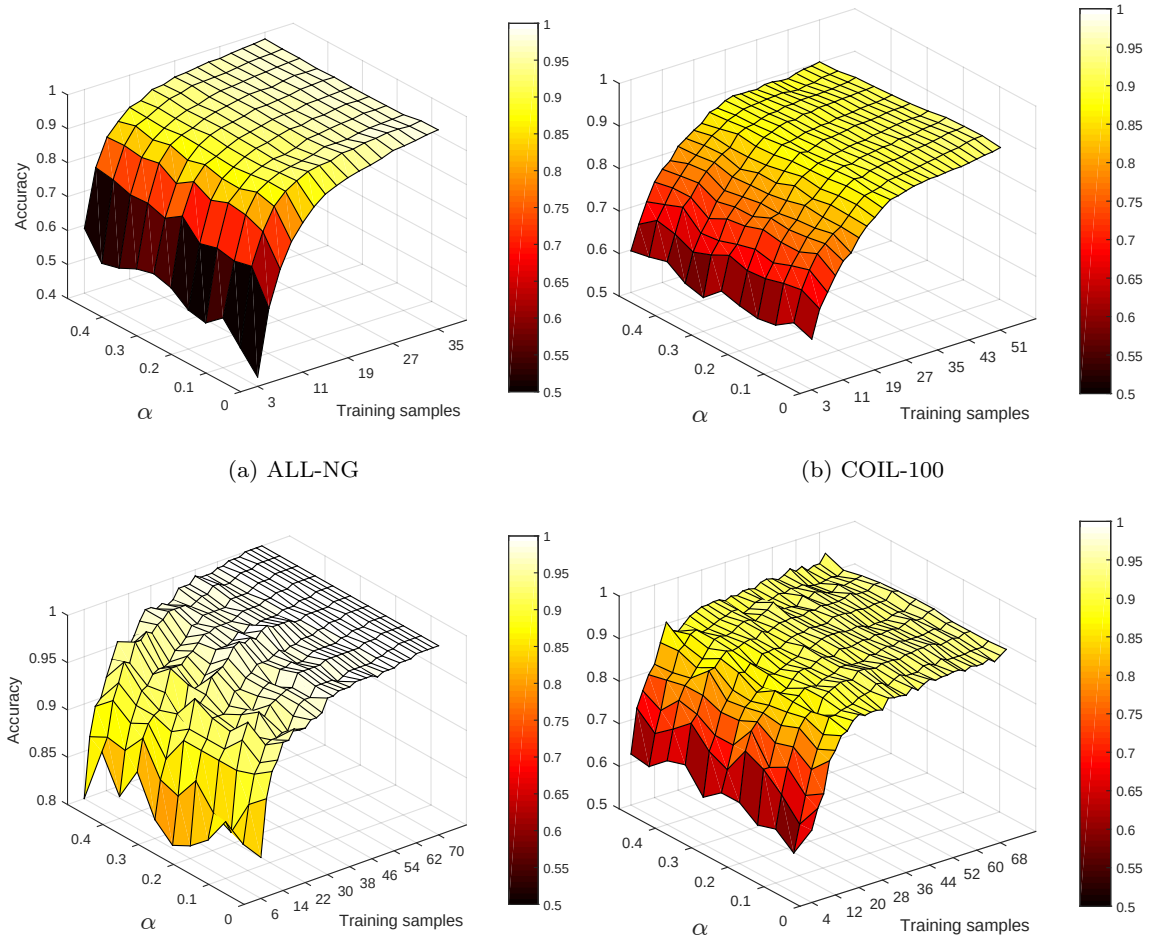
4.2.1 KGDCV performance analysis over both the training set size and the α value

Figure 4 presents the accuracy rate as a function of the number of training samples per class (x-axis) and the variance added (α) (y-axis). We can observe that, as expected in any classifier, the higher the number of training samples is, the better the accuracy rate of both KDCV ($\alpha = 0$) and KGDCV ($0 < \alpha < 1$) results. Both KGDCV-KT and KGDCV-NPT provided identical accuracy results. In addition, for a given number of training samples, the accuracy of KGDCV does not vary significantly when modifying the variance added to the final subspace. This gives and additional advantage to our methodology since it makes the parameter α easy to tune.

4.2.2 Comparative analysis in the SSS vicinity over α

Figure 5 depicts the comparison in accuracy for KGDCV -KT, KGDCV-NPT, DCV, GDCV and KDCV in all datasets. Both linear DCV and non-linear KDCV are presented by a single dot, since $\alpha = 0$ in these methods. Our KGDCV achieved the best performance of all methods in all cases, with both KGDCV-KT and KGDCV-NPT approaches given the exactly same accuracy value. From the results, it is observed, that non-linear kernel methods present better results than linear ones (KDCV > DCV, KGDCV > DCV). It can also be noticed that the better generalisation properties from the extended null space in GDCV and KGDCV is reflected on an improvement in accuracy in most datasets (GDCV > DCV, KGDCV $\geq$ KDCV), although this effect is more clearly exhibit in the linear versions for having more space for improving. As a result, we can conclude that KGDCV outperforms or obtain the same results than KDCV, DCV and GDCV. Finally, accuracy does not vary significantly with α, which makes its tuning easy and less sensitive than in GDCV.

Regarding the CPU time, Figure 6 shows several interesting observations. In spite of their good discriminative performance, KDCV and KGDCV-KT exhibit the largest computational time due to the Kernel Trick



(a) ALL-NG (b) COIL-100

(c) NIST (d) USPS

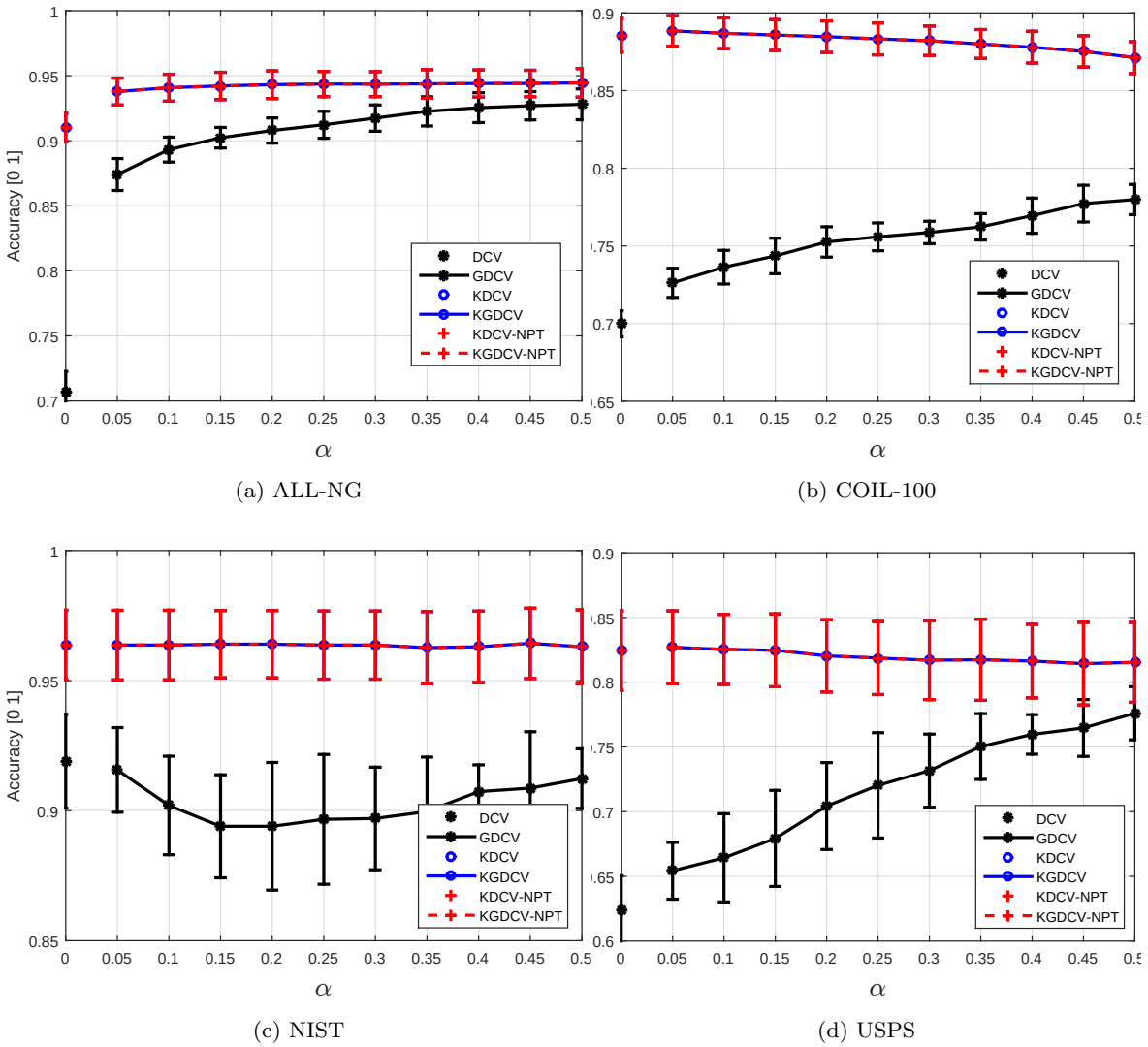
Fig. 4: Discriminative performance for KGDCV ($0 < \alpha < 1$) and KDCV ($\alpha = 0$) regarding the training set size and the α value.

implementation. However, our proposed KGDCV-NPT shows a drastic reduction on CPU time regarding the other non-linear methods, as expected from the discussion on computational complexity in sections 3.1 and 3.2, due to the different but more efficient computation of the re-mapping matrix. KGDCV-NPT also achieves a similar or even lower cost than linear methods as DCV and GDCV. Although this may seems contra-intuitive since KGDCV applies a GDCV as part of its algorithm -see Algorthm 2, step 5-, this is explained by the smaller size of the matrix Xo comparing to X. In general, non-generalized approaches are more expensive than the generalized ones (DCV > GDCV, KDCV > KGDCV) since the generalized approached reduces the dimensions of the range space involved in the calculations. Similarly to the accuracy analysis, the parameter α has a negligible effect on the CPU time, without any negative influence.

As conclusion of this experiment, our proposed KGDCV - NPT shows the best discriminative performance in terms of accuracy with the lowest computational time among all tested methods. It achieves a considerable computational gain without compromising the accuracy of the model regarding the KT approach.

4.2.3 Overall comparative analysis over the training size

Here we validate the accuracy rate and the CPU time of the both approaches, KGDCV by KT and NPT, regarding to the linear GDCV, KICA [17], KDAQR [27] and KDASR [3] methods, to a fixed α and an increasing training set size. The values of α are 0.9, 0.95, 0.95 and 0.85 to the ALL-NG, COIL-40/30, USPS and to NIST, respectively. In this final experiment, our proposed KGDCV is compared against other state-of-art



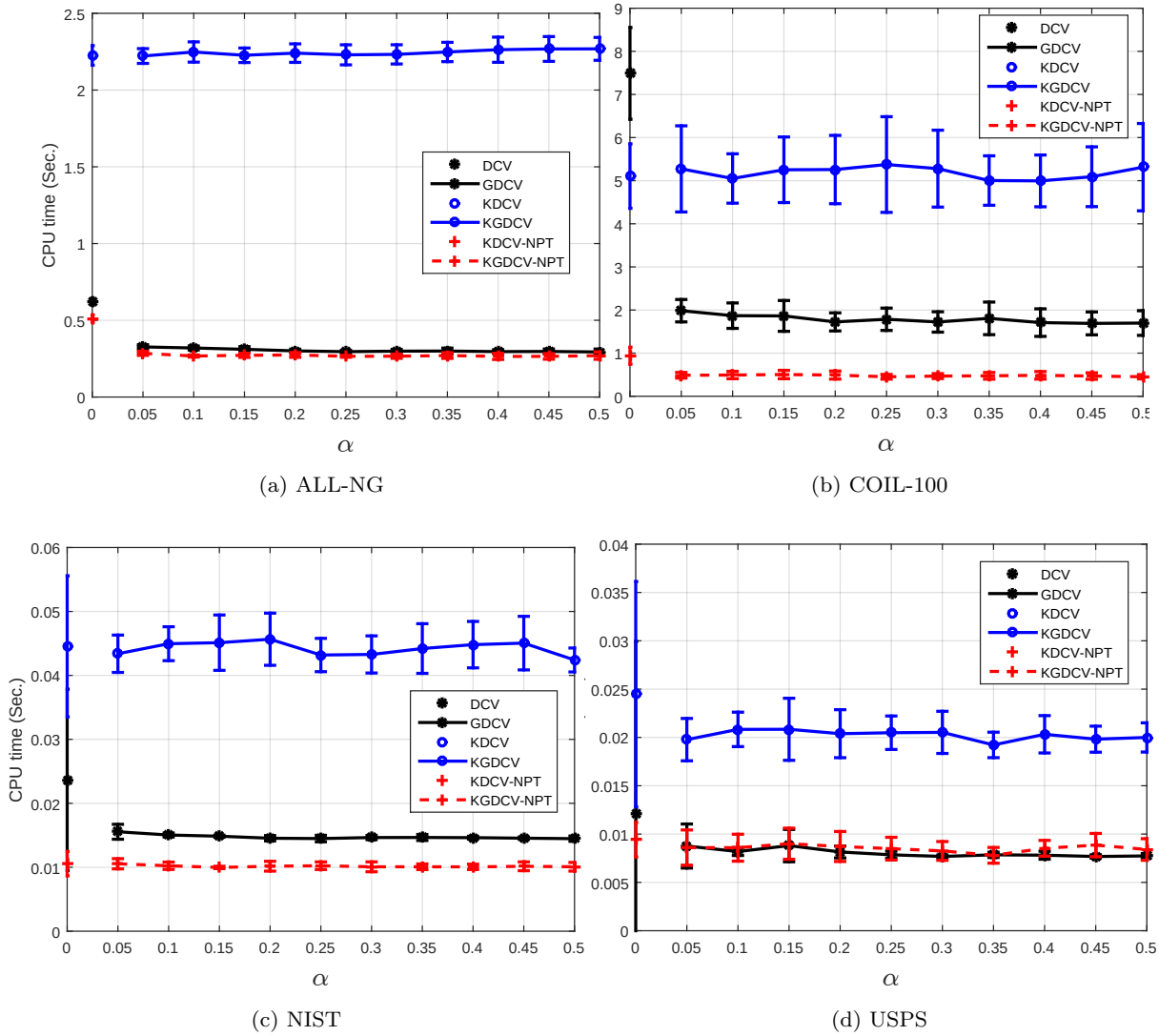
(a) ALL-NG (b) COIL-100

(c) NIST (d) USPS

Fig. 5: Discriminative performance comparison of accuracy over α between KGDCV-KT, KGDCV-NPT, DCV, GDCV and KDCV.

non-linear subspace learning methods based on different reductions. Linear GDCV is added as baseline to show the relative improvement achieved by non-linear techniques. Accuracy rates and CPU times of this comparison for all the four datasets are presented in Figures 7 and 8, respectively. The training set size is varied from the minimum possible to the maximum available in all datasets in order to validate the comparison under the most possible cases.

From these results, similar conclusions can be extracted as in the previous experiment. First, KGDCV achieves the best accuracy of all methods in the comparison for all datasets and training set sizes. Furthermore, the variant KGDCV-NPT also achieves the lowest computational cost of all non-linear methods. This difference is bigger as more training samples are available. As in Figure 6, KGDCV-NPT computational cost is also similar or lower than the linear GDCV, but only while $d > m$. For this reason, ALL-NG and USPS dataset, the biggest ones in number of samples with the smallest dimensionalities, exhibit a lower computation cost for GDCV from a certain amount of training samples onwards, due to the change in the relationship between d and m. KDASR is the second best method in the comparison both in time and accuracy, but the difference with KGDCV-NPT is clear. All non-linear approaches overperform GDCV in accuracy, but at the expense of computational cost, with the exception of our efficient KGDCV-NPT implementation.



(a) ALL-NG (b) COIL-100

(c) NIST (d) USPS

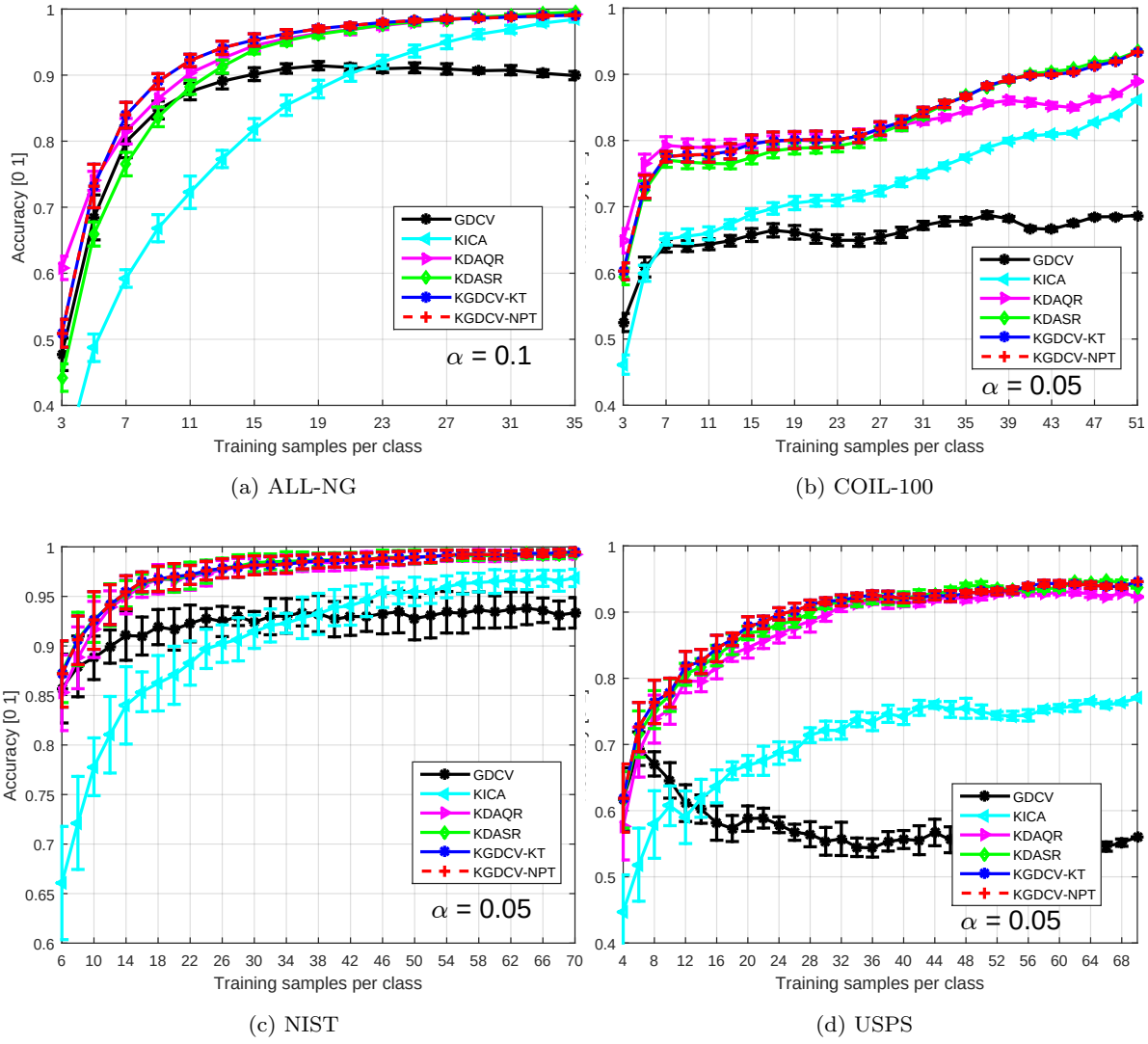
Fig. 6: Comparison of the CPU training time over α between KGDCV-KT, KGDCV-NPT, DCV, GDCV and KDCV.

5 Conclusions

In this paper, our method Kernel Generalized Discriminative Common Vectors (KGDCV) was presented as an approach to non-linear discriminant feature extraction. Our method combines the advantages of KDCV for non-linear spaces with the advantages of GDCV for better generalisation properties without restrictions on the training set size and lower computational complexity. Thus, KGDCV can be understood as a new extension of GDCV with kernels or as a novel generalization of KDCV.

Two different approaches to KGDCV were proposed, one based on the kernel trick (KT) and a second one based on the nonlinear projection trick (NPT) for higher efficiency.

Our method was validated on four different image datasets containing faces, objects and handwritten digits and compared against non-linear state-of-art methods as well as all the methods from which KGDCV is derived. In all tested cases, KGDCV approaches were the most discriminant methods in terms of accuracy. Moreover, our KGDCV-NPT showed simultaneously the best discriminative performance and the lowest computational time among all tested methods.



(a) ALL-NG (b) COIL-100

(c) NIST (d) USPS

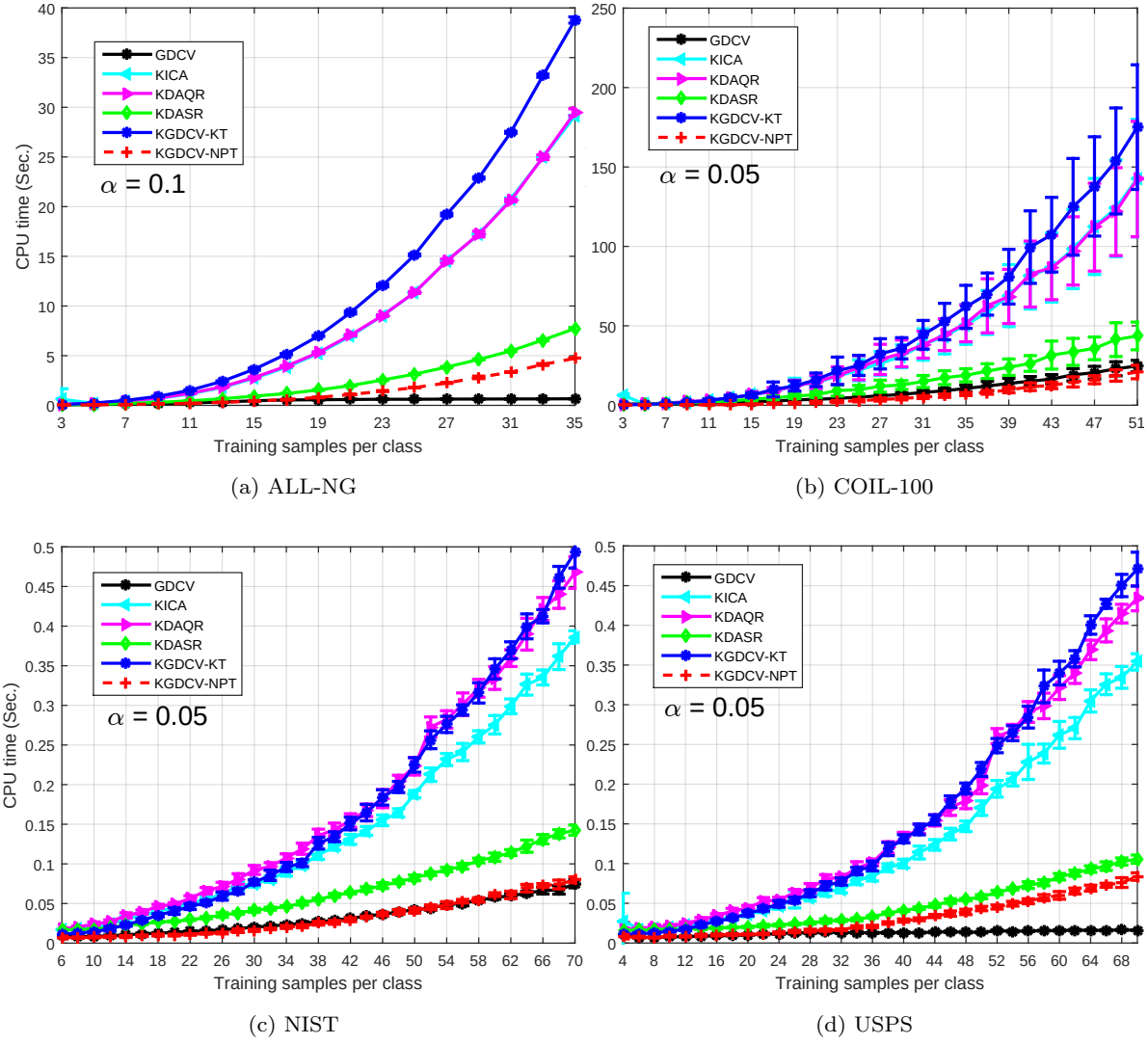
Fig. 7: Discriminative performance of accuracy over the training set size between KGDCV-KT, KGDCV-NPT, GDCV, KICA [17], KDAQR [27] and KDASR [3].

Appendix

A Table 1 presents the list of acronyms of the document.

References

1. Arlandis, J., Perez-Cortes, J.C., Llobet, R.: Handwritten character recognition using the continuous distance transformation. pp. 940–943 (2000)
2. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 711–720 (1997)
3. Cai, D., He, X., Han, J.: Speed up kernel discriminant analysis. The VLDB Journal **20**(1), 21–33 (2011)
4. Cevikalp, H., Neamtu, M., Barkana, A.: The kernel common vector method: A novel nonlinear subspace classifier for pattern recognition. IEEE Transactions on Systems, Man, and Cybernetics, Part B **37**(4), 937–951 (2007)
5. Cevikalp, H., Neamtu, M., Wilkes, M.: Discriminative common vector method with kernels. IEEE Transactions on Neural Networks **17**(6), 1550–1565 (2006)
6. Cevikalp, H., Neamtu, M., Wilkes, M., Barkana, A.: Discriminative common vectors for face recognition. IEEE Trans. Pattern Analysis and Machine Intelligence **27**(1), 4–13 (2005)
7. Chen, G., Tsai, W.: An incremental-learning-by-navigation approach to vision-based autonomous land vehicle guidance in indoor environments using vertical line information and multiweighted generalized hough transform technique. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on **28**(5), 740–748 (1998)
8. Chen, L.F., Liao, H.Y., Ko, M.T., Lin, J.C., Yu, G.J.: A new lda-based face recognition system which can solve the small sample size problem. Pattern Recognition **33**(10), 1713–1726 (2000)



(a) ALL-NG (b) COIL-100

(c) NIST (d) USPS

Fig. 8: Comparison of the CPU training time over the training set size between KGDCV-KT, KGDCV-NPT, GDCV, KICA [17], KDAQR [27] and KDASR [3].

9. Cun, Y.L., Boser, B., Denker, J.S., Howard, R.E., Habbard, W., Jackel, L.D., Henderson, D.: Advances in neural information processing systems 2. chap. Handwritten Digit Recognition with a Back-propagation Network, pp. 396–404 (1990)
10. Ferri, F., Diaz-Chito, K., Diaz-Villanueva, W.: Fast approximated discriminative common vectors using rank-one svd updates. In: Neural Information Processing, vol. 8228, pp. 368–375 (2013)
11. Fukunaga, K.: Introduction to Statistical Pattern Recognition, 2 edn. Academic Press (1990)
12. Georghiades, A., Belhumeur, P., Kriegman, D.: From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans. Pattern Anal. Mach. Intelligence **23**(6), 643–660 (2001)
13. van der Heijden, F., Duin, R.P.W., de Ridder, D., Tax, D.M.: Classification, Parameter Estimation and State Estimation: An Engineering Approach Using Matlab. Wiley (2004)
14. Howland, P., Wang, J., Park, H.: Solving the small sample size problem in face recognition using generalized discriminant analysis. Pattern Recognition **39**(2), 277–287 (2006)
15. Kovacs, Z., Guerrieri, R.: Computer recognition of handwritten characters using the distance transform. Electronics Letters **28**(19), 1825–1827 (1992)
16. Kwak, N.: Nonlinear projection trick in kernel methods: An alternative to the kernel trick. IEEE Transactions on Neural Networks and Learning Systems **24**(12), 2113–2119 (2013)
17. Liu, Q., Cheng, J., Lu, H., Ma, S.: Modeling face appearance with nonlinear independent component analysis. In: Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings., pp. 761–766 (2004)
18. Martinez, A., Benavente, R.: The ar face database. Technical Report 24, Computer Vision Center CVC (1998)
19. Nene, S., Nayar, S., Murase, H.: Columbia object image library (coil-100). Technical Report CUCS-006-96,

Table 1: List of acronyms.

KT	Kernel Trick
NPT	Nonlinear Projection Trick
SSS	Small Sample Size problem
EVD	Eigen-Value/Vector Decomposition
DCV	Discriminative Common Vectors
GDCV	Generalized Discriminative Common Vectors
KDCV	Kernel Discriminative Common Vectors
KGDCV	Kernel Generalized Discriminative Common Vectors
KICA	Kernel Independent Component Analysis
KDAQR	Kernel Discriminant Analysis by using QR decomposition
KDASR	Kernel Discriminant Analysis by using Spectral Regression
KDCV-KT	KDCV by using Kernel Trick
KGDCV-KT	KGDCV by using Kernel Trick
KDCV-NPT	KDCV by using Nonlinear Projection Trick
KGDCV-NPT	KGDCV by using Nonlinear Projection Trick

Department of Computer Science, Columbia University (1996)
20. Samaria, F., Harter, A.: Parameterisation of a stochastic model for human face identification. In: WACV94, pp. 138–142 (1994)
21. Schlkopf, B., Smola, A.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT Press (2002)
22. Schlkopf, B., Smola, A., Mller, K.: Nonlinear component analysis as a kernel eigenvalue problem. Technical Report 44, Max Planck Institute for Biological Cybernetics, Tbingen, Germany (1996)
23. Schlkopf, B., Smola, A., Mller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation **10**(5), 1299–1319 (1998)
24. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, New York, NY, USA (2004)
25. Tamura, A., Zhao, Q.: Rough common vector: A new approach to face recognition. In: IEEE Intl. Conf. on Syst, Man and Cybernetics,, pp. 2366–2371 (2007)
26. Wechsler, H., Phillips, P., Bruce, V., Fogelman, F., Huang, T.e.: Face recognition: From theory to applications. NATO ASI Series F, Computer and Systems Sciences **163**, 446–456 (1998)
27. Xiong, T., Ye, J., Li, Q., Cherkassky, V., Janardan, R.: Efficient kernel discriminant analysis via qr decomposition. In: Proceedings of the 17th International Conference on Neural Information Processing Systems, NIPS'04, pp. 1529–1536 (2004)
28. Yang, M.: Kernel eigenfaces vs. kernel fisherfaces: Face recognition using kernel methods. In: Proc. 5th Int. Conf.Automat. Face Gesture Recognition, pp. 215–220 (2002)
29. Zheng, J., Huang, Q., Chen, S., Wang, W.: Efficient kernel discriminative common vectors for classification. The Visual Computer **31**(5), 643–655 (2015)