Improved Image Discrimination Using Fast Non-linear Orthogonal Dictionary Learning

Improved Image Discrimination using Fast Non-linear Orthogonal Dictionary Learning

Puneet S Chhabra, Andrew M Wallace
School of Electrical and Physical Sciences
Heriot-Watt University, Edinburgh, EH144AS UK
Email: {psc31, a.m.wallace}@hw.ac.uk

James R Hopgood
IDCOM, School of Engineering
University of Edinburgh, Edinburgh, EH93FG
Email: james.hopgood@ed.ac.uk

***Abstract*—Most real-world signals or images have an intrinsic non-linear similarity measure and can be harder to discriminate. Kernel dictionary learning with applications to signal classification offers a solution to such a problem. However, decomposing a kernel matrix for large datasets is a computationally intensive task. Existing papers on dictionary learning using optimal kernel approximation method improve computation run-time but learn an over-complete dictionary. In this paper, we show that if we learn a discriminative orthogonal dictionary instead then learning and classification run-time can be significantly reduced. The proposed algorithm, Kernelized simultaneous approximation, and discrimination (K-SAD), learns a single highly discriminative and incoherent non-linear dictionary on small to medium-scale real-world datasets. Extensive experiments result in $> 97\%$ classification accuracy and show that the algorithm can scale both in space and time when compared to existing dictionary learning algorithms.**

***Index Terms*—Kernel, Dictionary learning, Sparsity.**

I. Introduction

Sparsity promotes a simple idea, a minimal collection of directions or atoms, called a *dictionary*, can represent a particular observation in the input or feature space. In most cases the underlying process that causes an observation to occur in the first place is low-dimensional. Identifying such a cause is highly beneficial for signal reconstruction, compression [1] and discrimination [2]. Dictionary learning (DL) for signal approximation [3] and discrimination [4, 5] is equal to identifying, given a set of training samples, an appropriate set or dictionary such that any K-subset of it spans a K-dimensional subspace. In contrast to hand-crafted dictionaries, DL methods adapt an over-complete or orthogonal dictionary to an observation, hoping for better sparsity.

A. Orthogonal or Over-complete Dictionary Learning

Given a measurement vector, $\mathbf{w} \in \mathbb{R}^{P\times 1} \subset \mathbf{W} \in \mathbb{R}^{P\times N}$, the aim is to extract a sparse vector, $\mathbf{q} \in \mathbb{R}^{K\times 1} \subset \mathbf{Q} \in \mathbb{R}^{K\times N}$, and learn a dictionary $\mathbf{Z} = [z_1, ..., z_K] \subset \mathbb{R}^{P\times K}$, simultaneously. Traditional DL algorithms have two steps:

1) **Sparse Coding**:

$$\min_{\mathbf{q_i}} ||\mathbf{w}_i - \mathbf{Z}\mathbf{q}_i||_2^2, \text{s.t.} \, ||\mathbf{q}_i||_0 \leqslant T_0, \forall i = 1, ..., N \quad (1)$$

2) **Dictionary Update**:

$$\min_{\mathbf{Z}} \left[||\mathbf{W} - \sum_{k=1}^{\mathrm{K}} \mathbf{z}_k \mathbf{q}_k^T||^2 \right], |z_k| = 1, \forall k = 1, ..., K \quad (2)$$

Equation (2) assumes that both $\mathbf{W}$ and $\mathbf{Z}$ are fixed except in column $\mathbf{z}_k$ and the coefficients that correspond to it, the k^{th} row in $\mathbf{Q}$, denoted as $\mathbf{q}_k^T$. $||.||_0$ is the sparsity measure. The unit length and orthogonal constraint makes the dictionary $\mathbf{Z}$ orthonormal.

Learning an over-complete dictionary using greedy methods ensures maximum sparsity, but the dictionary can be most coherent, i.e. highly redundant. Enforcing the incoherence condition on an over-complete dictionary [6] whilst solving 2 is a difficult task. In this work, we learn a structured orthogonal dictionary and use the derived coefficients as target signatures for classification purposes. Recently, Kernel based DL [7]–[9] methods have been proposed as an effective way of capturing non-linearity in the input space and learn sparse encodings, simultaneously. However, these methods require the kernel or Gram matrix, $\mathbf{K} \in \mathbb{R}^{N\times N}$. For large-scale datasets, computing such a matrix is a computationally complex task, both in space and time. Work by Golts and Elad [10] incorporate an effective way of approximating a kernel matrix with an over-complete dictionary and Gangeh *et. al* [11] show that an kernelized orthogonal dictionary can be learnt. However, in this paper, we combine the kernel approximation method [12] with a discriminative orthogonal dictionary learning step. This separates our work from [10] and [11] improving classification accuracy and reduces the algorithm run-time significantly.

B. Contributions & Outline

The key contributions of this work are: i) we report an improvement in run-time and classification accuracy on existing kernel DL methods [9, 10] by proposing to learn a discriminative orthogonal dictionary instead of an over-complete one; ii) unlike [10], we propose the use of an efficient SVD method for large matrices when approximating the kernel matrix using the Krylov method [12]; iii) we report state-of-the-art classification results and faster run-time on high-dimensional RGB-D and face recognition databases learning a single kernelised orthogonal dictionary; iv) finally, unlike [10, 11] we also map the kernel dictionary back into the input domain in order to better understand the dictionary structure and diversity.

ISBN 978-0-9928626-7-1 © EURASIP 2017

II. Kernel Dictionary Learning

The Mercer kernel defines an implicit, non-linear transformation mapping the input data into a higher or even an infinite dimensional kernel feature space [13]. The *kernel trick* allows training of the input data in the high dimensional feature space without explicitly computing the exact mapping. The Mercer kernel $\mathtt{k} : \mathsf{W} \times \mathsf{W} \mapsto \mathbb{R}$ for training samples w_i and w_j can be expressed as

$$\mathbf{K}_{i,j} = \mathtt{k}(w_i, w_j) = \langle \Phi(w_i), \Phi(w_j) \rangle, \; \forall i, j = 1, ..., N \quad (3)$$

where Φ is the implicit non-linear mapping associated with the kernel function $\mathtt{k}(\cdot, \cdot)$. For the input matrix $\mathbf{W} \in \mathbb{R}^{P \times N}$, the kernel matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ contains values of all pairs of input signals where $\Phi(w) \in \mathbb{R}^D$ is the image of w in F and $D \gg P$ is the dimension of the feature space F. Commonly used kernel methods are the linear kernel, polynomial kernels and Gaussian radial basis function (RBF). The RBF kernel can be expressed as

$$k(w_i, w_j) = \exp\left(-\gamma ||w_i - w_j||_2^2\right), \text{where}, \gamma > 0 \quad (4)$$

A. Kernelised Orthogonal Dictionary Learning Problem

We assume that for an input matrix $\mathbf{W} \in \mathbb{R}^{P \times N}$ its kernel matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ is of rank $r \leqslant N$. Hence, $\mathbf{K} \approx \mathbf{B}^T\mathbf{B} = \Phi(\mathbf{W})^T\Phi(\mathbf{W})$, where $\mathbf{B} \in \mathbb{R}^{r \times N}$. Finally using $\mathbf{B}$, we compute "virtual samples" $\mathbf{\Phi}_{train} \in \mathbb{R}^{K \times N}$, where $K \ll P$. Section III-A details how we approximate $\mathbf{\Phi}_{train}$.

Proposition 1. *Given the virtual samples, $\mathbf{\Phi}_{train}$, a dictionary, $\mathbf{Z} \in \mathbb{R}^{K \times K} \mid \mathbf{Z^T Z} = I$, where $K \ll P$, the original sparse coding* (1) *can be re-written as $J(\mathbf{Q}) =$:*

$$\min_{\mathbf{Q}} ||\mathbf{\Phi}_{train} - \mathbf{ZQ}||_2^2 + \beta_1 ||\mathbf{Q}||_1 + \beta_2 \mathbf{G}(\mathbf{Q}), \quad \textit{subject to} \; ||\mathbf{Q}||_1 \leqslant 1 \quad (5)$$

and has a unique solution $\mathbf{Q}^* = \mathrm{T}_{\beta_1}(\mathbf{Z}\,\mathbf{\Phi_{train}}, \mathbf{G}(\mathbf{Q}))$.

Proof. See Appendix A. □

We add a discriminatory function $\mathbf{G}(\mathbf{Q})$ (See Section III-B) that maximises inter-class variance and minimises intra-class variance of dictionary coefficients [2, 7]. We solve (5) using a soft-threshold operator $\mathrm{T}_{\beta_1}(\mathbf{Z}\,\mathbf{\Phi}_{train}) = sign(\mathbf{\Phi}_{train})\, max(|\mathbf{\Phi}_{train}| - \beta_1, 0)$.

Proposition 2. *Assuming $\mathbf{Z^T Z} = I$, the orthogonal kernel dictionary learning step can be written as:*

$$\min_{\mathbf{Z}} ||\mathbf{\Phi}_{train} - \mathbf{ZQ}||^2 \quad (6)$$

has a unique solution $\mathbf{Z}^* = \mathbf{U}\mathbf{V}^T$, where $\mathbf{U}$,$\mathbf{V}$ denote the orthogonal matrices defined by the following SVD $\mathbf{\Phi}_{trian}\mathbf{Q^T} = \mathbf{U\Sigma V}^T$.

Proof. See Appendix B. □

Algorithm 1: K-SAD

Input: $\mathbf{W}_{train}, \mathbf{W}_{test}$, sampler, sr, kernel, kt, c
Output: $labels$

begin
  `// Stage I - LKA. See Section III-A`
  $\mathbf{W}_s \mapsto VQ\left(\mathbf{W}_{train}, sr, kt, c\right)$
  $\mathbf{C}_{train} \mapsto compute\_kernel\left(\mathbf{W}_{train}, \mathbf{W}_s\right)$
  $\mathbf{C}_{test} \mapsto compute\_kernel\left(\mathbf{W}_{test}, \mathbf{W}_s\right)$
  $\mathbf{H} \mapsto compute\_kernel\left(\mathbf{W}_s, \mathbf{W}_s\right)$
  $\mathbf{H}^\dagger \mapsto \mathbf{\Lambda\Sigma}^\dagger\mathbf{\Lambda}^T$
  $\mathbf{\Phi}_{train} = \left(\mathbf{\Sigma}_k^\dagger\right)^{1/2} \mathbf{\Lambda}_k^T \mathbf{C}_{train}^T$
  $\mathbf{\Phi}_{test} = \left(\mathbf{\Sigma}_k^\dagger\right)^{1/2} \mathbf{\Lambda}_k^T \mathbf{C}_{test}^T$
  `// Stage 2 - ODL. Section III-B`
  Set initial $\mathbf{Z}_0$
  forall the $t \in [0, T]$ **do**
    $\mathbf{Q}_t = \mathrm{T}_{\beta_1}(\mathbf{Z}_t\mathbf{\Phi}_{train}, \mathbf{G}(\mathbf{Q}))$
    $\mathbf{\Phi}_{train}\mathbf{Q}_t^T = \mathbf{U\Sigma V}^T$
    $\mathbf{Z}_{t+1} = \mathbf{UV}^T$
  $\mathbf{Z} = \mathbf{Z}_{k+1}$
  `// Stage 3 - K-NN Classifier`
  $\mathbf{Q}_{test} = \mathbf{Z}^T\mathbf{\Phi}_{test}$
  $labels \leftarrow \mathrm{ModelKNNClassifier}(\mathbf{Q}_{test}, \mathbf{Z})$
  forall the $i \in [1, \mathrm{N}]$ **do**
    $\mathrm{Distance}(\mathbf{Z}, \mathbf{Q}_{test})$
  $labels \leftarrow \mathrm{Sort}(\mathrm{Distance}(\mathbf{Z}, \mathbf{Q}_{test}))$
return $labels$

III. Proposed Approach

Some of the limitations with linear and non-linear sparsity based classification algorithms are:

i) The Eigenvalue decomposition of the kernel or the Gram matrix may not scale with large data sets, $O(\mathrm{N}^2)$ and $O(\mathrm{N}^3)$ in space and time, respectively, where N is the number of observations.
ii) Learnt dictionaries may be highly redundant and may not have a structure.

Recent work by Golts and Elad [10] also propose a solution to the implicit kernel problem, i.e. efficient computation of the kernel matrix $\mathbf{K}$. However, unlike our approach, they do not enforce incoherency or discriminatory constraints on the dictionaries and learn over-complete dictionaries for each target class. Our algorithm learns only one.

A. Stage 1: Low-rank Kernel Approximation

The *low-rank kernel approximation* (LKA) stage is a pre-processing step that maps the high-dimensional input data on a low-dimensional non-linear feature space. We use the Nyström method, first introduced by Williams and Seeger [14] through uniform sampling of the input data. This method

ISBN 978-0-9928626-7-1 © EURASIP 2017

computes a low-rank approximation to $\mathbf{K}$ of the form $\tilde{\mathbf{K}} = \mathbf{C}\mathbf{H}_k^{\dagger}\mathbf{C}^T$. The Nyström method permutes $\mathbf{K}$ as:

$$\mathbf{C} = \begin{bmatrix} \mathbf{H} \\ \mathbf{S} \end{bmatrix} \mathbf{K} = \begin{bmatrix} \mathbf{H} & \mathbf{S^T} \\ \mathbf{S} & \mathbf{A} \end{bmatrix} \tag{7}$$

where $\mathbf{C}$ denotes the $\mathrm{N} \times c$ matrix formed by c columns, $\mathbf{H} \in \mathbb{R}^{c \times c}$ is a matrix consisting of the intersection of c columns with corresponding c rows of $\mathbf{K}$, $\mathbf{A}$ corresponds to a matrix composed of the remaining $(\mathrm{N} - c)$ rows and columns and $\mathbf{S} \in \mathbb{R}^{(\mathrm{N}-c) \times c}$ is a mixture of both. In this work, we use two sampling techniques that determine the size of c, i) random uniform sampling, and ii) vector quantisation (VQ) which uses the k-means clustering method. The uniform sampling method selects $c \ll \mathrm{N}$ columns from $\mathbf{W}_{train}$ at random compared to the clustering or the VQ method that uses c cluster centers. Finding the best sampling technique for LKA is out of the scope of this work and interested readers can see [14] and reference therein for a detailed analysis. Using (7) we construct $\tilde{\mathbf{K}}$ as follows $\tilde{\mathbf{K}} \approx \mathbf{C}\mathbf{H}^{\dagger}\mathbf{C}^T$, where $\left(\cdot^{\dagger}\right)$ denotes the pseudo-inverse operator.

Since $\mathbf{H}$ is a symmetric positive semi-definite (SPSD) matrix, it can also be written in terms of its eigenvalues and eigenvectors. Hence, we re-write $\mathbf{H}$ as

$$\mathbf{H} = \mathbf{\Lambda}\mathbf{\Sigma}\mathbf{\Lambda}^T \quad \text{and} \quad \mathbf{H}^{\dagger} = \mathbf{\Lambda}\mathbf{\Sigma}^{\dagger}\mathbf{\Lambda}^T \tag{8}$$

and $\left(\mathbf{H}^{\dagger}\right)^{1/2} = \left(\mathbf{\Sigma}^{\dagger}\right)^{1/2}\mathbf{\Lambda}^T$. Finally, we re-write $\mathbf{\Phi}_{train}$ as follows

$$\mathbf{\Phi}_{train} = \left(\mathbf{\Sigma}_{\mathbf{k}}^{\dagger}\right)^{1/2}\mathbf{\Lambda}_k^T\mathbf{C}^T \tag{9}$$

We solve Line 6 of algorithm 1 using the randomised version of the block Lanczos method [12] which is adapted for large datasets and produces nearly optimal accuracy. We repeat the above steps for the test dataset and get $\mathbf{\Phi}_{test}$.

B. Stage 2: Discriminative Coefficient based Orthogonal DL

Stage 2 of algorithm 1 presents the pseudocode of the ODL stage. The input to the algorithm is a training matrix $\mathbf{\Phi}_{train}$.The optimisation problem in (1) does not optimise the learned coefficients for maximum discrimination. The discriminative term in (5), $\mathrm{G}\left(\mathbf{Q}\right)$ is expressed below [2]. For a set of coefficients $\mathbf{Q} = [q_1, q_2, ..., q_{\mathrm{K}}]$, where $q_1, ..., q_k, ..., q_{\mathrm{K}}$ are the coefficients for the dictionary *atoms*, of which K_c samples are in class Ω_c, for $1 \leqslant c \leqslant \Omega$, the mean and variance for class Ω_c can be defined as: $\mu_c = \frac{1}{\mathrm{K}_c}\sum_{q \in \Omega_c} q$, and $\upsilon_c^2 = \frac{1}{\mathrm{K}_c}\sum_{z \in \Omega_c} ||z - \mu_c||_2^2$. The mean of all coefficient samples can be written as: $\mu = \frac{1}{\mathrm{K}}\sum_{k=1}^{\mathrm{K}} q_k$. The *inter-class* scatter matrix, S_w and the *intra-class* scatter matrix, S_b can be defined as: $S_b = || \sum_{c=1}^{\Omega} \mathrm{K}_c \left(\mu_c - \mu\right)\left(\mu_c - \mu\right)^T ||_2^2$ and $S_w = \sum_{c=1}^{\Omega} \upsilon_c^2$. Finally, the discrimination function is defined as $\mathrm{G}\left(\mathbf{Q}\right) = Trace(S_w^{-1}S_b)$. Algorithm 1 lists the pseudocode of the steps involved and we call this algorithm kernelised simultaneous approximation and discrimination (K-SAD).

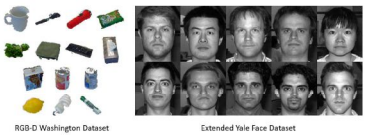

Fig. 1: RGB-D Washington and YaleB Face dataset.

IV. Experiments

The aim of our experiments is mainly to compare with other kernel dictionary learning approaches. We evaluate our approach on four publicly available benchmark datasets: i) The RGB-D object dataset [15], ii) *The ORL AT&T face dataset*[1] [16], iii) *Extended Yale face dataset*[2] [17] and iv) *MNIST Digit Dataset*[3] . All our experiments were carried out on an Intel quad-core i7-4800MQ 64-bit computer with a CPU clock speed of 2.7 GHz and 16 GB RAM.

A. RGB-D Object Recognition Dataset

The Washington RGB-D dataset is a collection of 300 household objects grouped into 51 categories collected using the Microsoft Kinnect sensor. The images are of size $\approx 85 \times 85 \times 4$. Several representation based methods, e.g. instance distance learning (IDL) [18], query adaptive similarity measure (QSM) [19], convolutional-recursive deep learning (CNN-RNN) [20], convolutional k-means descriptor (CKM) [21], depth kernel descriptors (KDES) [22] and hierarchical matching pursuit (HMP) [23] have reported results on the Washington RGB-D datasets. Lai *et. al* in [18] compute a single feature vector combining image, texture and depth features. Such features are then used for classification. Deep learning based methods, e.g. the CKD [20, 21], have reported state-of-the-art results where feature responses are learned in the vicinity of interest points and later combined into a descriptor. The CKD descriptor incorporates depth information which is then computed on image patches whose dimensions are pre-defined.

As suggested in [15] 10 trials with pre-defined training and test datasets[4] were adopted in our experiments and average accuracy is reported. We compare our results against state-of-the-art results in [19] and report classification accuracy and training and classification run-time. All the methods shown in Table I use the same training and test partitioning of the dataset. For our experiments we do not down-sample the images or extract any features [18, 22]; 51 dictionary atoms (one atom per category) are initialised at random from the input data and adopted using the DL method. A polynomial kernel of degree 8 with hyper-parameters $\beta_1 = 0.1$ and $\beta_2 = 0.01$ was used in our experiments.

B. The AT&T (formerly ORL) Face Dataset

The AT&T face dataset is composed of 40 subjects and 10 images with pose and expression variation per subject.

[1] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
[2] http://vision.ucsd.edu/~leekc/ExtYaleDatabase/
[3] http://yann.lecun.com/exdb/mnist/
[4] http://rgbd-dataset.cs.washington.edu/dataset/rgbd-dataset_eval/

ISBN 978-0-9928626-7-1 © EURASIP 2017

TABLE I: Accuracies(%) on RGB-D Washington dataset.

Method	Accuracy(%)
SP-HMP [23]	87.5 ± 2.9
IDL [18]	85.4 ± 3.2
CKM [21]	86.4 ± 2.3
CNN-RNN [20]	87.6 ± 2.0
KDES [22]	86.2 ± 2.1
Kernel SVM [18]	83.8 ± 3.5
QSM [19]	92.7 ± 1.0
Our approach	$\mathbf{97.572 \pm 0.265}$

	Method	Accuracy	Time(s)
AT&T Dataset	OMP [24]	93.75±2.12	-
	$L_1 - L_s$ [25]	95.90 ± 1.15	-
	KK-SVD [9]	93.75 ± 0.05	8.3
	FDDL [7]	94.25 ± 0.03	1744.3
	LKA+FDDL [7]	92.75 ± 0.03	92.78
	Our approach	$\mathbf{96.58 \pm 0.02}$	**1.3**
YaleB Dataset	OMP [24]	91.97±0.96	-
	$L_1 - L_s$ [25]	94.22 ± 0.71	-
	KK-SVD [9]	91.53 ± 0.09	41.96
	LKA+FDDL [7]	87.2 ± 2.12	9000
	FISTA [26]	94.50 ± 0.82	-
	PALM [27]	93.19 ± 0.642	-
	LKDL [10]	96.33	-
	Our approach	$\mathbf{98.26 \pm 0.03}$	**31.4**

TABLE II: Accuracies(%) on AT&T and YaleB Datasets

The dataset has images of size 112×92, captured on several different occasions in an up-right frontal position under a homogeneous background. We compare our approach to orthogonal matching pursuit [24], $L_1\_L_s$ [25] and FISTA [26]. Technical details of the above algorithms can be found in a recent survey [27]. In comparison to the methods discussed in a recent review [27] our method gives the least classification error. In our experiments, we use the full feature space of $10,304$ pixels as input to the stage 1 of our algorithm, unlike [27]. We randomly initialise 240 our dictionary atoms (6 per subject) from the non-linear mapped input data. A polynomial kernel of degree 8 was chosen for this experiment with $\beta_1 = 0.1$ and $\beta_2 = 0.01$. Table II illustrates state-of-the-art results on the AT&T face dataset.

C. Extended YaleB Face Dataset

The "Extended YaleB" face recognition database, in contrast to the AT&T database, is a larger database with 2,432 frontal images taken under varying lighting conditions and expressions. There are 38 subjects with ≈ 64 8-bit images per subject of size 192×168. Table II illustrates state-of-the-art results reported for this database against methods presented in [27]. In comparison to the methods and other kernel based DL methods our approach shows improvement in classification accuracy and is faster. For our experiments we do not re-size our images as done in [27] and use the full input space as an input to stage 1 of our algorithm. A polynomial kernel of degree 8 with 740 dictionary atoms (20 atoms per subject), initialised by uniformly sampling the kernel space was used.

TABLE III: Accuracies(%) on USPS Digit Dataset

Method	Accuracy(%)	Time(s)
FDDL [7]	95.79	-
FDDL+ LKDL [10]	96.03	-
LKA + KKSVD [9]	74.62	9825.4
SVM (Gaussian Kernel)	**98.6**	-
Our approach	**96.42**	**92.86**

D. MNIST USPS Digit Dataset

The USPS MNIST dataset consists of $60,000$ training image and $10,000$ test images of size 28×28. The parameters used for this experiment are: $\beta_1 = 0.1, \beta_2 = 0.01$, 20 DL iterations, 300 dictionary atoms and a polynomial kernel of order 8. We compare classification accuracies and run-time execution against state-of-the-art results reported by Golts *et. al* [10]. Table III compares classification accuracy of our approach against approaches presented in [9, 10]. We use the KKSVD code made available by the authors. The original algorithm in [9] is not feasible for such a large dataset hence we employ the kernel approximation (Stage 1, III-A) algorithm first and then use the KKSVD algorithm for kernel dictionary learning. We call this method "LKA + KKSVD" in Table III. Compared to [10], with same parameters, the orthogonal discriminatory dictionary learnt using our approach is ≈ 55 times faster.

V. Conclusion

This work improves on two central problems in DL algorithms i) handle non-linearity in the input space by improving classification accuracy on existing publicly available datasets; ii) further reducing the learning and classification algorithm run-time through kernel matrix approximation. This work combines an orthogonal incoherent discriminatory dictionary learning method in the non-linear space with an efficient approximation of kernel matrix. Unlike some existing techniques, which have produced these ideas individually, our approach learns a single non-linear orthogonal dictionary which is incoherent, minimising cardinality and maximising the discrimination capabilities in the non-linear space. We complement the small-runtime required by the orthogonal DL step with a fast kernel approximation stage in our algorithm. We report state-of-the-art results on large-scale high-dimensional datasets and report an average classification accuracy of $\approx 97\%$ on 8-bit digits, face and RGB-D images. Unlike most sparsity based classifiers our approach uses the coefficients as target signatures. Finally, unlike [10, 11], the reverse mapping, *i.e.* pre-images, of the kernel dictionary, Figure 2, were computed using [28].

Acknowledgment

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/K014277/1 and the MOD University Defence Research Collaboration in Signal Processing.

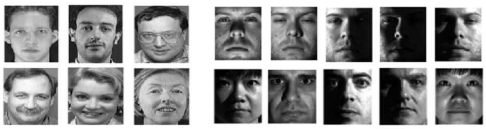

Fig. 2: Exemplars of the learnt kernel dictionary.

APPENDIX A
PROOF FOR PROPOSITION 1

We apply the majorisation-minimisation method to solve our non-linear cost function $J(\mathbf{Q})$ (5) using a surrogate function $M(\mathbf{Q},\mathbf{Q}_t) = \frac{1}{2}(\mathbf{Q}-\mathbf{Q}_t)^T(\alpha\mathbf{I}-\mathbf{Z}^T\mathbf{Z})(\mathbf{Q}-\mathbf{Q}_t)$. Hence by design $\hat{J}(\mathbf{Q}) = J(\mathbf{Q}) + M(\mathbf{Q},\mathbf{Q}_t)$ coincides with $J(\mathbf{Q})$ at $\mathbf{Q}_t$. Solving the modified cost function leads to

$$\hat{J}(\mathbf{Q}) = \mathbf{\Phi}_{train}^T\mathbf{\Phi}_{train} - 2\mathbf{\Phi}_{train}^T\mathbf{Z}\mathbf{Q} + \mathbf{Q}^T\mathbf{Z}^T\mathbf{Z}\mathbf{Q} + (\mathbf{Q}-\mathbf{Q}_t)^T(\alpha\mathbf{I}-\mathbf{Z}^T\mathbf{Z})(\mathbf{Q}-\mathbf{Q}_t) \quad (10)$$

$$\frac{\partial J(\mathbf{Q})}{\partial \mathbf{Q}} = -2\mathbf{Z}^T\mathbf{\Phi}_{train} - 2(\alpha\mathbf{I}-\mathbf{Z}^T\mathbf{Z})\mathbf{Q}_t + 2\beta_1\mathbf{Q} = 0 \Rightarrow \mathbf{Q}^* = \mathrm{T}_{\beta_1}\big(\mathbf{Q}_t + \frac{1}{\alpha}\mathbf{Z}^T\mathbf{\Phi}_{train} - \mathbf{Z}\mathbf{Q}_t + G(\mathbf{Q}_t)\big), \frac{\beta_1}{2\alpha}\big) \quad (11)$$

APPENDIX B
PROOF FOR PROPOSITION 2

The reduced rank procrustes rotation (Theorem 4 in [29]) shows that for the minimisation problem for $\mathbf{Z}$ in

$$\min_{\mathbf{Z}} ||\mathbf{\Phi}_{train} - \mathbf{Z}\mathbf{Q}||^2 \quad (12)$$

subject to $\mathbf{Z}^T\mathbf{Z} = \mathbf{I}$, has a unique solution $\mathbf{Z}^* = \mathbf{U}\mathbf{V}^T$, where $\Phi_{train}\mathbf{Q}^T = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$. In contrast to this work, where the dictionary learnt is a kernel dictionary, the proof in [29] is in the linear domain.

REFERENCES

[1] O. Bryt and M. Elad, "Compression of facial images using the k-svd algorithm," *Journal of Visual Communication and Image Representation*, vol. 19, no. 4, pp. 270–282, 2008.

[2] P. S. Chhabra, A. Maccarone, A. McCarthy, A. M. Wallace, and G. S. Buller, "Discriminating underwater lidar target signatures using sparse multi-spectral depth codes," in *Sensor Signal Processing for Defence (SSPD)*, IEEE, 2016.

[3] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *Signal Processing, IEEE Transactions on*, vol. 54, no. 11, pp. 4311–4322, 2006.

[4] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 791–804, 2012.

[5] Q. Zhang and B. Li, "Discriminative k-svd for dictionary learning in face recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 2691–2698, IEEE, 2010.

[6] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.

[7] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *2011 International Conference on Computer Vision*, pp. 543–550, IEEE, 2011.

[8] Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent k-svd: Learning a discriminative dictionary for recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651–2664, 2013.

[9] H. Van Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Kernel dictionary learning," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2021–2024, IEEE, 2012.

[10] A. Golts and M. Elad, "Linearized kernel dictionary learning," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 726–739, 2016.

[11] M. J. Gangeh, A. Ghodsi, and M. S. Kamel, "Kernelized supervised dictionary learning," *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4753–4767, 2013.

[12] N. Halko, P.-G. Martinsson, Y. Shkolnisky, and M. Tygert, "An algorithm for the principal component analysis of large data sets," *SIAM Journal on Scientific computing*, vol. 33, no. 5, pp. 2580–2594, 2011.

[13] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American mathematical society*, vol. 68, no. 3, pp. 337–404, 1950.

[14] C. Williams and M. Seeger, "Using the nyström method to speed up kernel machines," in *Proceedings of the 14th annual conference on neural information processing systems*, no. EPFL-CONF-161322, pp. 682–688, 2001.

[15] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 1817–1824, IEEE, 2011.

[16] F. S. Samaria and A. C. Harter, "Parameterisation of a stochastic model for human face identification," in *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, pp. 138–142, IEEE, 1994.

[17] K.-C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 5, pp. 684–698, 2005.

[18] K. Lai, L. Bo, X. Ren, and D. Fox, "Sparse distance learning for object recognition combining rgb and depth information," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 4007–4013, IEEE, 2011.

[19] Y. Cheng, R. Cai, C. Zhang, Z. Li, X. Zhao, K. Huang, and Y. Rui, "Query adaptive similarity measure for rgb-d object recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 145–153, 2015.

[20] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3d object classification," in *Advances in Neural Information Processing Systems*, pp. 665–673, 2012.

[21] M. Blum, J. T. Springenberg, J. Wülfing, and M. Riedmiller, "A learned feature descriptor for object recognition in rgb-d data," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 1298–1303, IEEE, 2012.

[22] L. Bo, X. Ren, and D. Fox, "Depth kernel descriptors for object recognition," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 821–826, IEEE, 2011.

[23] L. Bo, X. Ren, and D. Fox, "Unsupervised feature learning for rgb-d based object recognition," in *Experimental Robotics*, pp. 387–402, Springer, 2013.

[24] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation. part i: Greedy pursuit," *Signal Processing*, vol. 86, no. 3, pp. 572–588, 2006.

[25] K. Koh, S.-J. Kim, and S. Boyd, "An interior-point method for large-scale l1-regularized logistic regression," *Journal of Machine learning research*, vol. 8, no. Jul, pp. 1519–1555, 2007.

[26] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM journal on imaging sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[27] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, "A survey of sparse representation: algorithms and applications," *IEEE Access*, vol. 3, pp. 490–530, 2015.

[28] J.-Y. Kwok and I.-H. Tsang, "The pre-image problem in kernel methods," *IEEE transactions on neural networks*, vol. 15, no. 6, pp. 1517–1525, 2004.

[29] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Journal of computational and graphical statistics*, vol. 15, no. 2, pp. 265–286, 2006.

ISBN 978-0-9928626-7-1 © EURASIP 2017