Pattern Recognition 75 (2018) 282–291

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Nonlinear dictionary learning with application to image classification

Junlin Hu*, Yap-Peng Tan

School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798, Singapore

ARTICLE INFO

Article history:
Received 11 August 2016
Revised 29 December 2016
Accepted 3 February 2017
Available online 4 February 2017

Keywords:
Nonlinear dictionary learning
Sparse coding
Neural network
Image classification

ABSTRACT

In this paper, we propose a new nonlinear dictionary learning (NDL) method and apply it to image classification. While a variety of dictionary learning algorithms have been proposed in recent years, most of them learn only a linear dictionary for feature learning and encoding, which cannot exploit the nonlinear relationship of image samples for feature extraction. Even though kernel-based dictionary learning methods can address this limitation, they still suffer from the scalability problem. Unlike existing dictionary learning methods, our NDL employs a feed-forward neural network to seek hierarchical feature projection matrices and dictionary simultaneously, so that the nonlinear structure of samples can be well exploited for feature learning and encoding. To better exploit the discriminative information, we extend the NDL into supervised NDL (SNDL) by learning a class-specific dictionary with the labels of training samples. Experimental results on four image datasets show the effectiveness of the proposed methods.

1. Introduction

In recent years, sparse representation has been widely studied in signal processing and machine learning [1,2], and it has also been successfully applied to various computer vision applications such as image denoising [3], face recognition [4–7], facial analysis [8–11], image classification [12–15], and visual tracking [16]. The basic assumption of sparse representation is that one signal can be well approximated by a linear combination of a small number of atoms (or basis) from an over-complete dictionary. Generally, the dictionary plays an important role in sparse modeling, and its quality can heavily affect the performance of sparse representation [17]. Therefore, dictionary learning is a basic element to sparse representation. Instead of using a predefined dictionary (e.g., various wavelets), recent advances in dictionary learning have shown that learning a desirable dictionary from the training data itself can usually yield good results for numerous image and video analysis tasks [3,12,13,18–24].

There have been a number of dictionary learning methods proposed in recent years [2,19–21,25]. These dictionary learning methods can be roughly divided into two categories: *unsupervised* and *supervised*. Unsupervised dictionary learning methods aim to learn an over-complete dictionary by minimizing the reconstruction error of a set of signals with sparsity constraints [13,20,26], which have shown good performance in some visual reconstruction and clustering tasks [3,22] (e.g., image denoising, image restoration, and video clustering). For the second category, supervised dictionary learning methods usually learn a compact and discriminative dictionary by exploiting the label or side information of the training data, which complements a discriminative term to the reconstruction error and optimizes the objective functions for different settings [18,19,21,27–30]. Methods in this category are usually applied to various visual recognition tasks [13,25,31]. To promote the discriminative capacity of the learned dictionary, there are several strategies that can be used for this discriminative term such as introducing a classifier on sparse coefficients [19,21,27], learning a structured dictionary using the incoherence constraint between class-specific dictionaries [22,32] and the Fisher discrimination criterion [23] on sparse coefficients. Generally, the dictionary learned by supervised methods can achieve good performance for many visual applications [21,31].

Most existing dictionary learning algorithms, however, usually learn a linear dictionary for feature learning and encoding in the original space such that they cannot capture the nonlinear structure of data points. To address this nonlinearity problem, the kernel trick is often adopted to map the data points into a high-dimensional space and then learn a dictionary in this transformed space using existing dictionary learning approaches [25,33]. However, the kernel-based methods cannot explicitly obtain the nonlinear mapping function and often suffer from the scalability problem. Different from these methods, in this paper, we propose a nonlinear dictionary learning (NDL) method to seek hierarchical nonlinear projection matrices and dictionary simultaneously via a feed-forward neural network, so that the nonlinear structure of samples can be well exploited. To better exploit the discriminative

* Corresponding author.
E-mail addresses: jhu007@e.ntu.edu.sg (J. Hu), eyptan@ntu.edu.sg (Y.-P. Tan).

http://dx.doi.org/10.1016/j.patcog.2017.02.009

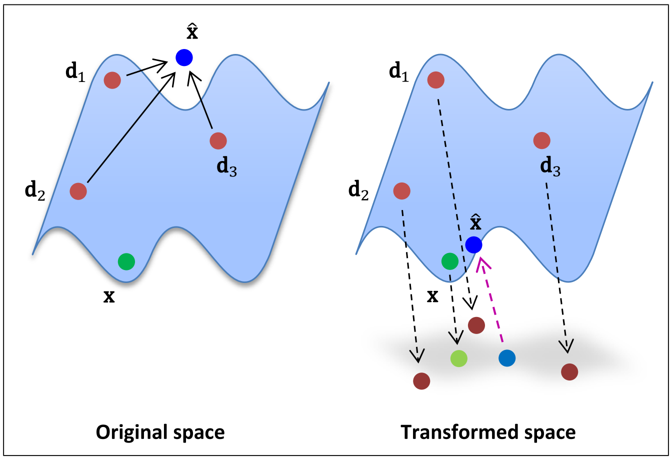


Fig. 1. A toy example illustrates how NDL captures the nonlinearity of data points. In the **original** space, $\widehat{\mathbf{x}}$ (blue point) is the linear approximation of the data point $\mathbf{x}$ (green) over three atoms $\mathbf{d}_1$, $\mathbf{d}_2$ and $\mathbf{d}_3$ of a dictionary (red points), and it is not on the manifold $\mathcal{M}$. The NDL method maps both the $\mathbf{x}$ and atoms to a new space via a feed-forward neural network and then computes the linear combination on this **transformed** space so that the representation $\widehat{\mathbf{x}}$ of the $\mathbf{x}$ is on the manifold $\mathcal{M}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

information, we extend the NDL into supervised NDL (SNDL) by learning a class-specific dictionary with the label of training samples. Fig. 1 shows a toy example illustrating how the proposed NDL captures the nonlinearity of data points.

2. Related work

2.1. Deep learning

Deep learning has been a popular research topic in the communities of machine learning and computer vision due to its excellent performance in various visual tasks such as face recognition [34–42], image classification [43,44], action recognition [45] and visual tracking [46]. A variety of deep learning methods have been proposed to directly learn rich hierarchical feature representations from raw data [47]. Representative deep learning models are deep belief networks [48], deep convolutional neural networks (CNN) [44], deep stacked auto-encoder [45], and deep metric network (or siamese network) [49–53]. However, to our best knowledge, little attempt has been made on coordinating deep learning and dictionary leaning to utilize the merits of deep learning. In this work, we introduce a nonlinear dictionary learning approach to learn several hierarchical nonlinear transformations and the desired dictionary simultaneously by integrating deep learning and dictionary leaning into a unified framework.

2.2. Dictionary learning

Learning a desired dictionary from the training data for sparse representation has attracted considerable attention in the field of computer vision [2,21], and many advanced dictionary learning algorithms have been proposed in recent years. The K-SVD method [26] is a representative unsupervised dictionary learning approach, which learns an over-complete dictionary from natural images in an iterative fashion. Based on K-SVD, a discriminative term is usually added to the reconstruction error to obtain discriminative dictionaries [19,21,23,54–56]. For example, Mairal et al. [21,27] proposed to learn a shared dictionary for all classes and discriminative class models on sparse coefficients simultaneously. Jiang et al. [19] introduced the label consistent K-SVD algorithm to balance reconstructive and discriminative power of the learned dictionary. To capture the nonlinearity of data, kernel-based dictionary learning approaches [25,33] employ kernel trick to first map the data from the original space into another space and then utilize well-known dictionary learning methods in this space. More recently, deep sparse coding (DeepSC) [57] extends sparse coding to a multi-layer architecture, where it employed an approach called dimensionality reduction by learning an invariant mapping (DrLIM) [49] to learn a parametric mapping in the siamese network. Different from the DeepSC and hierarchical sparse coding [58] methods which treat the output of sparse coding as the input of the next layer, our NDL method employs a feedforward neural network to learn a nonlinear mapping and dictionary simultaneously, so that the nonlinear structure of data points can be well exploited.

3. Nonlinear dictionary learning

We briefly introduce some background of the dictionary learning methods, and then focus on formulating nonlinear dictionary learning method in an unsupervised manner.

3.1. Background

Let $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_K] \in \mathbb{R}^{r\times K}$ be a dictionary that contains K atoms. In the classical sparse coding task, a signal $\mathbf{x} \in \mathbb{R}^r$ can be sparsely represented by a linear combination of a few atoms from the dictionary $\mathbf{D}$ as:

$$\mathbf{x} \approx \mathbf{D}\ \mathbf{a} = a_1\mathbf{d}_1 + a_2\mathbf{d}_2 + \cdots + a_K\mathbf{d}_K, \tag{1}$$

where $\mathbf{a} = [a_1, a_2, \cdots, a_K]^T \in \mathbb{R}^K$ is a sparse coefficient vector. The sparse coding with a ℓ_1 regularization problem is usually solved to obtain an optimal sparse solution $\mathbf{a}$:

$$\min_{\mathbf{a}} \quad \|\mathbf{x} - \mathbf{D}\ \mathbf{a}\|_2^2 + \lambda_1\|\mathbf{a}\|_1, \tag{2}$$

where $\|\mathbf{x}\|_2$ and $\|\mathbf{x}\|_1$ denote ℓ_2 and ℓ_1 norms of vector $\mathbf{x}$ respectively, and λ_1 is a positive regularization parameter. The first term in (2) is the reconstruction error, and the second term is the sparsity penalty.

To learn a dictionary from the training set of N samples $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in \mathbb{R}^{r\times N}$, dictionary learning (DL) algorithms [20,21] simply minimize the following empirical cost function over both a dictionary $\mathbf{D}$ and a sparse matrix $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_N] \in \mathbb{R}^{K\times N}$ as:

$$\begin{aligned}&\min_{\mathbf{D},\ \mathbf{A}} \quad \frac{1}{N}\sum_{i=1}^{N}\left(\|\mathbf{x}_i - \mathbf{D}\ \mathbf{a}_i\|_2^2 + \lambda_1\|\mathbf{a}_i\|_1\right)\\ &\text{s.t.} \quad \|\mathbf{d}_i\|_2^2 \le 1, \ \ \forall\ i = 1, 2, \cdots, K,\end{aligned} \tag{3}$$

where vector $\mathbf{a}_i$ is the ith column of $\mathbf{A}$ and the sparse coefficient of sample $\mathbf{x}_i$ over $\mathbf{D}$. The constraint $\{\|\mathbf{d}_i\|_2^2 \le 1\}_{i=1}^K$ aims to prevent $\mathbf{D}$ from being arbitrarily large because it would cause very small values of the $\mathbf{A}$. The problem (3) can be solved by several works such as the feature-sign search algorithm [20] and gradient descent method.

3.2. Nonlinear dictionary learning

Unlike most existing dictionary learning methods which only learn a linear dictionary, our NDL employs a feed-forward neural network to exploit the nonlinearity of samples for dictionary learning. Fig. 2 illustrates the basic idea of the proposed NDL method.

Consider a feed-forward neural network with 2M+1 layers, which includes two parts: encoding and decoding, and there are $r^{(m)}$ neurons in layer m, holding $r^{(2M-m)} = r^{(m)}$ for all $m = 0, 1, \cdots, 2M$. The representation $\mathbf{h}^{(m)}$ of an input $\mathbf{x} \in \mathbb{R}^{r^{(0)}}$ in the layer m $(m \ge 1)$ is represented as:

$$\mathbf{h}^{(m)} = \varphi(\mathbf{z}^{(m)}) \in \mathbb{R}^{r^{(m)}}, \tag{4}$$

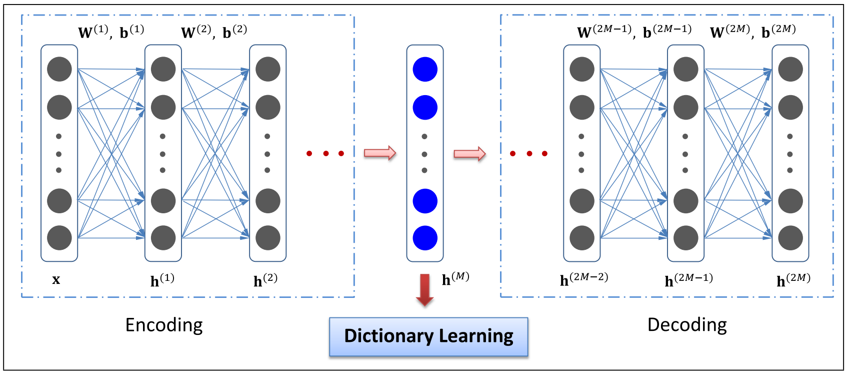


Fig. 2. The basic idea of our NDL method. Considering a feedforward neural network with $2M+1$ layers, $\mathbf{h}^{(0)} = \mathbf{x}$ is the input to this network, $\mathbf{h}^{(m)}$ is the output of the layer m, and $\mathbf{W}^{(m)}$ and $\mathbf{b}^{(m)}$ are the parameters of the network to be learned, $1 \le m \le 2M$. In encoding part, the resulting output is $\mathbf{h}^{(M)}$, and we perform dictionary learning on the layer M by using some optimization criterions. In decoding part, the final output is $\widehat{\mathbf{x}} = \mathbf{h}^{(2M)}$, which is the reconstruction of the input $\mathbf{x}$. Finally, the back-propagation algorithm is used to update the parameters of the network.

$$\mathbf{z}^{(m)} = \mathbf{W}^{(m)}\mathbf{h}^{(m-1)} + \mathbf{b}^{(m)}, \tag{5}$$

where $\mathbf{W}^{(m)} \in \mathbb{R}^{r^{(m)} \times r^{(m-1)}}$ and $\mathbf{b}^{(m)} \in \mathbb{R}^{r^{(m)}}$ are weight matrix and bias vector of this layer; and $\varphi : \mathbb{R} \mapsto \mathbb{R}$ is a nonlinear activation function (e.g., tanh and sigmoid); and $\mathbf{h}^{(0)} = \mathbf{x}$ denotes the original input. In the encoding part, the resulting representation of $\mathbf{x}$ at the layer M is denoted as $f_M(\mathbf{x}) = \mathbf{h}^{(M)} \in \mathbb{R}^{r^{(M)}}$, where the mapping $f_M :\ \mathbb{R}^{r^{(0)}} \mapsto \mathbb{R}^{r^{(M)}}$ is a parametric nonlinear function determined by $\{\mathbf{W}^{(m)},\ \mathbf{b}^{(m)}\}_{m=1}^{M}$. In the decoding part, the final output of $\mathbf{x}$ at the layer $2M$ is given as $\widehat{\mathbf{x}} = f_{2M}(\mathbf{x}) = \mathbf{h}^{(2M)} \in \mathbb{R}^{r^{(2M)}}$, which is the reconstruction of the $\mathbf{x}$, where $f_{2M} :\ \mathbb{R}^{r^{(0)}} \mapsto \mathbb{R}^{r^{(2M)}}$ is a nonlinear mapping parametrized by $\{\mathbf{W}^{(m)},\ \mathbf{b}^{(m)}\}_{m=1}^{2M}$.

Considering the mapping f_M, we cast the nonlinear dictionary learning problem on the manifold $\mathcal{M}$ as the conventional dictionary learning problem in the transformed space (i.e., layer M). In addition, to avoid producing meaningless results (e.g., uniform output) and preserve some structures of the original data, it is desired that the output $\widehat{\mathbf{x}} = f_{2M}(\mathbf{x})$ at the layer $2M$ should be the reconstruction of the original input $\mathbf{x}$. Based on these two points, the nonlinear dictionary learning (NDL) is formulated as the following optimization problem:

$$\begin{aligned}\min_{f_M,\ f_{2M},\ \mathbf{D},\ \mathbf{A}}\ \mathcal{J} &= \mathcal{J}_1 \ +\ \lambda_2\ \mathcal{J}_2\ +\ \lambda_3\ \mathcal{J}_3 \\ &= \frac{1}{N}\sum_{i=1}^{N}\Big(\|f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i\|_2^2 + \lambda_1\|\mathbf{a}_i\|_1\Big) \\ &\quad + \lambda_2\ \frac{1}{N}\sum_{i=1}^{N}\|f_{2M}(\mathbf{x}_i) - \mathbf{x}_i\|_2^2 \\ &\quad + \lambda_3\ \sum_{m=1}^{2M}\big(\|\mathbf{W}^{(m)}\|_F^2 + \|\mathbf{b}^{(m)}\|_2^2\big) \\ \text{s.t.}\quad & \|\mathbf{d}_i\|_2^2 \le 1,\ \ \forall\ i = 1, 2, \cdots, K,\end{aligned} \tag{6}$$

where $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_K] \in \mathbb{R}^{r^{(M)} \times K}$ is the dictionary learned in the layer M; $\mathbf{a}_i$ is the sparse vector corresponding to the sample $f_M(\mathbf{x}_i)$ and $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_N] \in \mathbb{R}^{K \times N}$ is the matrix form of all $\{\mathbf{a}_i\}_{i=1}^{N}$; and $\|\mathbf{X}\|_F$ represents the Frobenius norm of the matrix $\mathbf{X}$.

In (6), $\mathcal{J}_1$ is the conventional dictionary learning algorithm in the layer M, $\mathcal{J}_2$ is an autoencoder term which minimizes the reconstruction error between $f_{2M}(\mathbf{x}_i)$ and $\mathbf{x}_i$, where $\lambda_2 > 0$ balances the importance of the term $\mathcal{J}_2$ to the objective function $\mathcal{J}$, the term $\mathcal{J}_3$ is a regularization term to decrease the magnitude of the weights and help prevent overfitting in model training, and λ_3 is a positive regularization parameter to control the relative importance of this term to the objective function.

3.3. Optimization

The optimization problem in (6) is not jointly convex with respect to the mappings f_M, f_{2M}, dictionary $\mathbf{D}$ and sparse matrix $\mathbf{A}$, and it is non-trivial to obtain a global optimal solution. To solve this, we use an iterative method which alternately optimizes the f_M, f_{2M}, $\mathbf{D}$ and $\mathbf{A}$.

Stage 1 (Nonlinear Mapping): Optimize f_{M}, $\mathrm{f}_{2\mathrm{M}}$ *with the fixed* $\mathbf{D}$ *and* $\mathbf{A}$. The problem (6) can be rewritten as following optimization when both $\mathbf{D}$ and $\mathbf{A}$ are fixed:

$$\begin{aligned}\min_{f_M,\ f_{2M}}\ \mathcal{J} &= \frac{1}{N}\sum_{i=1}^{N}\|f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i\|_2^2\ +\ C_1 \\ &\quad + \lambda_2\ \frac{1}{N}\sum_{i=1}^{N}\|f_{2M}(\mathbf{x}_i) - \mathbf{x}_i\|_2^2 \\ &\quad + \lambda_3\ \sum_{m=1}^{2M}\big(\|\mathbf{W}^{(m)}\|_F^2 + \|\mathbf{b}^{(m)}\|_2^2\big),\end{aligned} \tag{7}$$

where C_1 is a constant.

The back-propagation algorithm is employed in conjunction with batch gradient descent method to update the parameters $\{\mathbf{W}^{(m)},\ \mathbf{b}^{(m)}\}_{m=1}^{2M}$. The partial derivatives of the $\mathcal{J}$ regarding the $\mathbf{W}^{(m)}$ and $\mathbf{b}^{(m)}$ are given as:

$$\begin{aligned}\frac{\partial \mathcal{J}}{\partial \mathbf{W}^{(m)}} &= u[M-m]\ \frac{2}{N}\sum_{i=1}^{N}\Delta_{M,i}^{(m)}\ \mathbf{h}_i^{(m-1)^T} \\ &\quad + \lambda_2\ \frac{2}{N}\sum_{i=1}^{N}\Delta_{2M,i}^{(m)}\ \mathbf{h}_i^{(m-1)^T}\ +\ 2\lambda_3\mathbf{W}^{(m)},\end{aligned} \tag{8}$$

$$\frac{\partial \mathcal{J}}{\partial \mathbf{b}^{(m)}} = u[M-m]\ \frac{2}{N}\sum_{i=1}^{N}\Delta_{M,i}^{(m)} + \lambda_2\ \frac{2}{N}\sum_{i=1}^{N}\Delta_{2M,i}^{(m)}\ +\ 2\lambda_3\mathbf{b}^{(m)}, \tag{9}$$

where the $\Delta_{M,i}^{(m)}$ is computed by:

$$\Delta_{M,i}^{(M)} = (f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i) \odot \varphi'(\mathbf{z}_i^{(M)}), \tag{10}$$

$$\Delta_{M,i}^{(m)} = \Big(\mathbf{W}^{(m+1)^T}\Delta_{M,i}^{(m+1)}\Big) \odot \varphi'\Big(\mathbf{z}_i^{(m)}\Big), \tag{11}$$

for layer $m = 1, 2, \cdots, M-1$, and $\Delta_{2M,i}^{(m)}$ is given by:

$$\Delta_{2M,i}^{(2M)} = (f_{2M}(\mathbf{x}_i) - \mathbf{x}_i) \odot \varphi'\big(\mathbf{z}_i^{(2M)}\big), \tag{12}$$

$$\Delta_{2M,i}^{(m)} = \Big(\mathbf{W}^{(m+1)^T} \Delta_{2M,i}^{(m+1)}\Big) \odot \varphi'\big(\mathbf{z}_i^{(m)}\big), \tag{13}$$

for the layer $m = 1, 2, \cdots, 2M-1$; and the operation $\odot$ denotes the element-wise multiplication. The $u[M-m]$ is the unit step function of discrete variable m, and it holds that $u[M-m] = 1$ for $m \leq M$, $u[M-m] = 0$ otherwise.

Then, the $\mathbf{W}^{(m)}$ and $\mathbf{b}^{(m)}$ can be updated by

$$\mathbf{W}^{(m)} = \mathbf{W}^{(m)} - \beta_1 \frac{\partial \mathcal{J}}{\partial \mathbf{W}^{(m)}}, \tag{14}$$

$$\mathbf{b}^{(m)} = \mathbf{b}^{(m)} - \beta_1 \frac{\partial \mathcal{J}}{\partial \mathbf{b}^{(m)}}, \tag{15}$$

and β_1 is the learning rate controlling the convergence speed of the objective function $\mathcal{J}$.

Stage 2 (Dictionary Update): Update $\mathbf{D}$ *using the learned* f_M *and* $\mathbf{A}$. When we fix the f_M and $\mathbf{A}$, the optimization problem in (6) over $\mathbf{D}$ is convex and it can be rewritten as:

$$\begin{aligned} \min_{\mathbf{D}} \quad & \mathcal{J} = \frac{1}{N}\sum_{i=1}^{N} \|f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i\|_2^2 \ + \ C_2 \\ \text{s.t.} \quad & \|\mathbf{d}_i\|_2^2 \leq 1, \ \forall \ i = 1, 2, \cdots, K, \end{aligned} \tag{16}$$

where C_2 is a constant. This is a least squares problem with quadratic constraints, and it can be efficiently solved by Lagrange multiplier method as used in [20]. We first construct the Lagrangian as follows:

$$\mathcal{L}(\mathbf{D},\ \boldsymbol{\mu}) = \frac{1}{N}\sum_{i=1}^{N} \|f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i\|_2^2 + \sum_{i=1}^{K} \mu_i\big(\|\mathbf{d}_i\|_2^2 - 1\big), \tag{17}$$

where the $\boldsymbol{\mu} = [\mu_1, \mu_2, \cdots, \mu_K]^T$ are Lagrange multipliers and each $\mu_i \geq 0$.

Let $\partial\mathcal{L}(\mathbf{D},\ \boldsymbol{\mu})/\partial\mathbf{D} = \mathbf{0}$, we can compute the analytical solution as:

$$\mathbf{D} = f_M(\mathbf{X})\mathbf{A}^T(\mathbf{A}\mathbf{A}^T + \boldsymbol{\Sigma})^{-1}, \tag{18}$$

where $f_M(\mathbf{X}) = [f_M(\mathbf{x}_1), \cdots, f_M(\mathbf{x}_N)] \in \mathbb{R}^{r^{(M)} \times N}$ and diagonal matrix $\boldsymbol{\Sigma} = N\ \mathrm{diag}(\boldsymbol{\mu}) \in \mathbb{R}^{K \times K}$. Hence, the corresponding Lagrange dual function is:

$$\begin{aligned} \mathcal{L}_{dual}(\boldsymbol{\mu}) &= \min_{\mathbf{D}} \quad \mathcal{L}(\mathbf{D},\ \boldsymbol{\mu}) \\ &= \frac{1}{N}\sum_{i=1}^{N} \|f_M(\mathbf{x}_i) - f_M(\mathbf{X})\mathbf{A}^T(\mathbf{A}\mathbf{A}^T + \boldsymbol{\Sigma})^{-1}\ \mathbf{a}_i\|_2^2 \\ &\quad + \sum_{i=1}^{K} \mu_i\big(\|f_M(\mathbf{X})\mathbf{A}^T(\mathbf{A}\mathbf{A}^T + \boldsymbol{\Sigma})^{-1}\ \mathbf{e}_i\|_2^2 - 1\big), \end{aligned} \tag{19}$$

where $\mathbf{e}_i \in \mathbb{R}^K$ is the i-th unit vector. This dual problem (19) can be solved by maximizing the Lagrange dual function $\mathcal{L}_{dual}(\boldsymbol{\mu})$ over the variables $\{\mu_i \geq 0\}_{i=1}^K$ using the gradient descent method. The gradient of the $\mathcal{L}_{dual}(\boldsymbol{\mu})$ with regard to the μ_i is calculated as:

$$\frac{\partial \mathcal{L}_{dual}(\boldsymbol{\mu})}{\partial \mu_i} = \|f_M(\mathbf{X})\mathbf{A}^T(\mathbf{A}\mathbf{A}^T + \boldsymbol{\Sigma})^{-1}\ \mathbf{e}_i\|_2^2 - 1. \tag{20}$$

Substituting the optimal $\boldsymbol{\mu}$ (or $\boldsymbol{\Sigma}$) into (18), we easily obtain the learned dictionary $\mathbf{D}$.

Stage 3 (Sparse Coding): Obtain $\mathbf{A}$ *with the learned* f_M *and* $\mathbf{D}$. Fixing f_M and $\mathbf{D}$, the optimization problem (6) with respect to each $\mathbf{a}_i$ is also convex, and it can be expressed as:

$$\min_{\mathbf{a}_i} \quad \mathcal{J} = \|f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i\|_2^2 + \lambda_1\|\mathbf{a}_i\|_1 \ + \ C_3, \tag{21}$$

where C_3 is a constant. This is a ℓ_1-regularized least squares problem, which is non-smooth because of involving the ℓ_1 norm term. To address this non-smooth problem, the epsilon-ℓ_1 norm is adopted to replace ℓ_1 norm following [20,59,60]. The epsilon-ℓ_1 norm of a vector $\mathbf{x}$ is $\sum_i (x_i^2 + \epsilon)^{\frac{1}{2}}$, where ϵ is a small positive constant (e.g., 10^{-6}). If $\epsilon \to 0$, then $\sum_i (x_i^2 + \epsilon)^{\frac{1}{2}}$ (epsilon-ℓ_1 norm) reduces to $\|\mathbf{x}\|_1 = \sum_i |x_i|$ (ℓ_1 norm).

Then, the derivative of the problem (21) over $\mathbf{a}_i$ can be calculated as:

$$\frac{\partial \mathcal{J}}{\partial \mathbf{a}_i} = -2\ \mathbf{D}^T(f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i) + \lambda_1\boldsymbol{\Sigma}_1\ \mathbf{a}_i, \tag{22}$$

where $\boldsymbol{\Sigma}_1 \in \mathbb{R}^{K \times K}$ is a diagonal matrix with its diagonal entries $\sigma_{jj} = (a_{ij}^2 + \epsilon)^{-\frac{1}{2}}$, $j = 1, 2, \cdots, K$. By setting the $\partial\mathcal{J}/\partial\mathbf{a}_i$ as 0, we can obtain the analytical solution:

$$\mathbf{a}_i = \left(\mathbf{D}^T\mathbf{D} + \frac{\lambda_1}{2}\boldsymbol{\Sigma}_1\right)^{-1} \mathbf{D}^T f_M(\mathbf{x}_i). \tag{23}$$

In practice, if a matrix $\mathbf{X}$ is singular, we use the $\mathbf{X} + \epsilon\ \mathbf{I}$ to compute its inverse, in which ϵ is a small positive constant and $\mathbf{I}$ is an identity matrix.

As $\boldsymbol{\Sigma}_1$ is dependent on $\mathbf{a}_i$ in (23), we can obtain $\mathbf{a}_i$ using the iterative method by alternatively optimizing $\mathbf{a}_i$ and $\boldsymbol{\Sigma}_1$. In practice, we initialize $\mathbf{a}_i$ as $(\mathbf{D}^T\mathbf{D} + \lambda_1\mathbf{I})^{-1}\mathbf{D}^T f_M(\mathbf{x}_i)$, which is the solution of the optimization problem:

$$\min_{\mathbf{a}_i} \quad \|f_M(\mathbf{x}_i) - \mathbf{D}\ \mathbf{a}_i\|_2^2 + \lambda_1\|\mathbf{a}_i\|_2^2, \tag{24}$$

where $\mathbf{I} \in \mathbb{R}^{K \times K}$ is an identity matrix.

We repeat the above three stages until the algorithm satisfies a certain convergence condition. Algorithm 1 summarizes the optimization procedure of the NDL method.

4. Supervised nonlinear dictionary learning

The NDL is an unsupervised learning approach, which doesn't utilize any label information of training samples. Generally, learning a discriminative dictionary by exploiting label information of samples can help improve the performance of various visual recognition tasks [19,27,55]. To this end, we propose a supervised nonlinear dictionary learning (SNDL) method to learn a class-specific dictionary for each class with the incoherence constraint.

Given a set samples $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_C]$ from C classes, where $\mathbf{X}_c = [\mathbf{x}_1^c, \mathbf{x}_2^c, \cdots, \mathbf{x}_{N_c}^c]$ is a collection of N_c samples from the c-th class, $c \in \{1, 2,..., C\}$, the optimization problem of the SNDL can be formulated as:

$$\begin{aligned} &\min_{f_M,\ f_{2M},\ \{\mathbf{D}_c,\ \mathbf{A}_c\}_{c=1}^C} \quad \mathcal{J} \\ &= \sum_{c=1}^{C}\left\{\frac{1}{N_c}\sum_{i=1}^{N_c}\Big(\|f_M(\mathbf{x}_i^c) - \mathbf{D}_c\ \mathbf{a}_i^c\|_2^2 + \lambda_1\|\mathbf{a}_i^c\|_1\Big)\right. \\ &\quad \left. + \ \lambda_4 \sum_{j=1,\ j\neq c}^{C} \|\mathbf{D}_c^T\mathbf{D}_j\|_F^2\right\} \\ &\quad + \ \lambda_2\sum_{c=1}^{C}\frac{1}{N_c}\sum_{i=1}^{N_c}\|f_{2M}(\mathbf{x}_i^c) - \mathbf{x}_i^c\|_2^2 + \lambda_3\sum_{m=1}^{M}\Big(\|\mathbf{W}^{(m)}\|_F^2 + \|\mathbf{b}^{(m)}\|_2^2\Big) \\ &\text{s.t.} \quad \|\mathbf{d}_i^c\|_2^2 \leq 1, \ \forall\ 1 \leq c \leq C, \ 1 \leq i \leq K_c, \end{aligned} \tag{25}$$

Algorithm 1: NDL.

```
Input: Training data: $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$; Parameters: $2M$ (number of layers
       of network), $\{r^{(m)}\}_{m=0}^{2M}$ (number of neurons), $K$ (size of the dictionary),
       $\lambda_1$, $\lambda_2$, $\lambda_3$, $\beta_1$ (learning rate), and $\varepsilon = 10^{-5}$ (convergence error).
Output: Dictionary $\mathbf{D}$, weights $\{\mathbf{W}^{(m)}\}_{m=1}^{2M}$, and biases $\{\mathbf{b}^{(m)}\}_{m=1}^{2M}$.
Initialization: Set $\mathbf{b}^{(m)} = \mathbf{0}$ and randomly initialize $\mathbf{W}^{(m)}$, $\mathbf{D}$ and $\mathbf{A}$ for all
$m = 1, 2, \cdots, 2M$; set $t = 1$ and $\mathcal{J}^0 = 0$; compute $\mathcal{J}^1$ via (6).
while not converge (i.e., $|\mathcal{J}^t - \mathcal{J}^{t-1}| > \varepsilon$) do
    stage 1 (nonlinear mapping): update $f_M$, $f_{2M}$ with the fixed $\mathbf{D}$ and $\mathbf{A}$:
    for $m = 2M, M-1, \cdots, 1$ do
        Obtain $\partial\mathcal{J}/\partial\mathbf{W}^{(m)}$ and $\partial\mathcal{J}/\partial\mathbf{b}^{(m)}$ via (8) and (9) by
        back-propagation;
    end
    for $m = 1, 2, \cdots, 2M$ do
        $\mathbf{W}^{(m)} \longleftarrow \mathbf{W}^{(m)} - \beta_1\ \partial\mathcal{J}/\partial\mathbf{W}^{(m)}$;
        $\mathbf{b}^{(m)} \longleftarrow \mathbf{b}^{(m)} - \beta_1\ \partial\mathcal{J}/\partial\mathbf{b}^{(m)}$;
    end
    stage 2 (dictionary update): update $\mathbf{D}$ with the learned $f_M$ and $\mathbf{A}$:
    Initialize $\boldsymbol{\mu} = \mathbf{1}$, and $\boldsymbol{\Sigma} = N\,\mathrm{diag}(\boldsymbol{\mu}) \in \mathbb{R}^{K\times K}$;
    while $\mathcal{L}_{dual}(\boldsymbol{\mu})$ (19) not converge do
        Update $\boldsymbol{\mu}$ using the gradient (20) by gradient descent method;
    end
    Obtain $\mathbf{D} = f_M(\mathbf{X})\mathbf{A}^T(\mathbf{A}\mathbf{A}^T + \boldsymbol{\Sigma})^{-1}$ as in (18);
    stage 3 (sparse coding): obtain $\mathbf{A}$ with the learned $f_M$ and $\mathbf{D}$:
    for $i = 1, 2, \cdots, N$ do
        Initialize $\mathbf{a}_i = (\mathbf{D}^T\mathbf{D} + \lambda_1\mathbf{I})^{-1}\mathbf{D}^T f_M(\mathbf{x}_i)$;
        while problem (21) not converge do
            Alternatively optimize $\boldsymbol{\Sigma}_1$ and $\mathbf{a}_i$ in (23);
        end
    end
    Obtain sparse matrix $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_N]$;
    $t = t + 1$;
    Compute $\mathcal{J}^t$ via (6);
end
```

where $\mathbf{D}_c = [\mathbf{d}_1^c, \cdots, \mathbf{d}_{K_c}^c]$ is the dictionary corresponding to the class c, and K_c is the number of atoms in the dictionary $\mathbf{D}_c$; and $\mathbf{A}_c = [\mathbf{a}_1^c, \cdots, \mathbf{a}_{N_c}^c]$ is the sparse matrix of $\mathbf{X}_c$ over the $\mathbf{D}_c$. The $\|\mathbf{D}_c^T\mathbf{D}_j\|_F^2$ is a cross-dictionary incoherence term, which promotes incoherence between two dictionaries $\mathbf{D}_c$ and $\mathbf{D}_j$, $c \neq j$, and λ_3 controls the importance of the incoherence term.

To solve this problem (25), we adopt the similar optimization procedure as used in the NDL (6). Specifically, in the **stage 2** (updating $\mathbf{D}_c$ with the learned f_M and $\{\mathbf{A}_c\}_{c=1}^C$), the optimization problem (25) with respect to the $\mathbf{D}_c$ is rewritten as:

$$\min_{\mathbf{D}_c} \mathcal{J} = \frac{1}{N_c}\sum_{i=1}^{N_c} \|f_M(\mathbf{x}_i^c) - \mathbf{D}_c\,\mathbf{a}_i^c\|_2^2 + \lambda_4 \sum_{j\neq c}^{C} \|\mathbf{D}_c^T\mathbf{D}_j\|_F^2$$
$$\text{s.t.}\quad \|\mathbf{d}_i^c\|_2^2 \leq 1,\ \ \forall\ i = 1, 2, \cdots, K_c. \tag{26}$$

We use a gradient descent algorithm to solve this problem (26), and the derivative of $\mathcal{J}$ with regard to the class-specific dictionary $\mathbf{D}_c$ is computed as:

$$\frac{\partial\mathcal{J}}{\partial\mathbf{D}_c} = \mathbf{D}_c\mathbf{A}_c\mathbf{A}_c^T - f_M(\mathbf{X}_c)\mathbf{A}_c^T + \lambda_4\left(\sum_{j\neq c}^{C}\mathbf{D}_j\mathbf{D}_j^T\right)\mathbf{D}_c. \tag{27}$$

Finally, the complete or shared dictionary can be given as $\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_C]$. We use this dictionary $\mathbf{D}$ to encode a sample $\mathbf{x}$ by solving the sparse coding problem in the transformed space f_M (i.e., the layer M):

$$\min_{\mathbf{a}}\ \ \|f_M(\mathbf{x}) - \mathbf{D}\,\mathbf{a}\|_2^2 + \lambda_1\|\mathbf{a}\|_1. \tag{28}$$

Additional optimization details of the SNDL follow similar procedures of those in NDL method that is summarized in Algorithm 1.

5. Experiments

In this section, we apply the proposed NDL and SNDL methods for image classification task, and evaluate their performance on four image categorization benchmarks.

5.1. Application to image classification

We adopted the bag-of-visual-words (BoV) and spatial pyramid matching (SPM) [61] framework for image classification. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$ be a set of feature descriptors (e.g., SIFT) from an image (or image patch), and let $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_N]$ be the corresponding sparse presentations of the $\mathbf{X}$ over the learned dictionary $\mathbf{D}$. The max pooling function was chosen to map the $\mathbf{A}$ into a single feature vector $\mathbf{s} \in \mathbb{R}^K$ by:

$$s_j = \max\left\{|a_{1j}|, |a_{2j}|, \cdots, |a_{Nj}|\right\}, \tag{29}$$

where s_j and a_{ij} are the jth element of vector $\mathbf{s}$ and $\mathbf{a}_i$, respectively. Following [13,61], we partitioned each entire image into 21 ($=1^2 + 2^2 + 4^2$) image patches at three scales using spatial pyramid representation. Then we performed max pooling on all the patches and concatenated them into a long feature vector of size 21 × K, which was finally fed into a multi-class linear SVM [13] for classification.

5.2. Implementation details

5.2.1. Datasets

We implemented and evaluated the proposed NDL and SNDL methods on four widely used image classification datasets detailed as follows:

- Scene-15 [61]: This dataset contains 4485 images falling into 15 categories, where the number of images for each category varies between 200 and 400. Following the standard settings on this dataset, we randomly took 100 images per category for training and used the rest images for testing. We repeated this procedure 5 times and reported the mean and standard deviation of classification accuracies.
- UIUC-Sports [62]: It contains 1579 images of eight sport event classes such as badminton, croquet, polo, rock climbing, sailing, and so on. We adopted 5 random splits of this dataset and recorded the mean accuracy and standard deviation, where 70 images per class were used for training and 60 for testing.
- MIT67 Indoor [63]: This dataset totally contains 15,620 images of 67 indoor scene categories. In the standard evaluation procedure, the training and test images were fixed, and 80 images of each class were used for training and 20 images for testing.
- Caltech-256 [64]: It is a more challenging dataset, containing 30,607 images in 256 categories with more variabilities. Following the common setting [64], we randomly selected 15 and 30 images per category for training, and the remaining images were used for testing. We partitioned this dataset five times and reported the mean accuracy with standard deviation.

Fig. 3 shows sample images from these four datasets. In experiments, we converted all the images into grayscale and resized each image no more than 300 pixels in each direction with the original aspect ratio for feature extraction. For each image, we used a single descriptor type and densely sampled 128-dimensional SIFT on image patch of size 16 × 16 pixels with 8-pixel spacing. To well exploit the spatial information, we adopted three-level spatial pyramid representation for each image as suggested in [13,61].

5.2.2. Parameter settings

We used a five-layer feedforward neural network ($M = 2$), where the number of neurons was set as 128 → 128 → 128 → 128 → 128 from the input to output layer. To exploit nonlinearity of data, we simply utilized the *tanh* as the nonlinear activation function for each layer. The learning rate β_1 was slowly reduced from an initial value of 0.1 in the optimization procedure. The parameters λ_1, λ_2, λ_3 and λ_4 were empirically set as 0.15, 1, 0.001 and 0.1 respectively. In regard to the dictionary size, we set $K = 1024$ for

(a) Scene-15

(b) UIUC-Sports

(c) MIT67 Indoor

(d) Caltech-256

Fig. 3. Sampled images from different datasets, where each three images, from top to bottom, are from the Scene-15, UIUC-Sports, MIT67 Indoor, and Caltech-256 datasets, respectively.

Table 1
The comparison of classification accuracy (%) with baselines on three scene datasets. The bold in Table denotes the best classification accuracy.

Method	Scene-15	UIUC-Sports	MIT67
ScSPM	80.13 ± 0.56	82.56 ± 1.01	38.77
K-ScSPM	81.15 ± 0.70	83.31 ± 0.80	39.51
NDL	82.75 ± 0.53	84.62 ± 0.68	41.56
SNDL	**83.69** ± **0.45**	**85.20** ± **0.63**	**42.03**

Table 2
The comparison of classification accuracy (%) with several methods on three scene datasets. The bold in Table denotes the best classification accuracy.

Method	Scene-15	UIUC-Sports	MIT67
ScSPM [13]	80.28 ± 0.93	–	–
DeepSC [57]	82.71 ± 0.98	–	–
Gist [63]	–	–	26.50
OB [65]	80.90	76.30	37.60
NDL	82.75 ± 0.53	84.62 ± 0.68	41.56
SNDL	**83.69 ± 0.45**	**85.20 ± 0.63**	**42.03**

Table 3
The comparison of classification accuracy (%) on the Caltech-256 dataset. The bold in Table denotes the best classification accuracy.

# training images	15	30
ScSPM	27.65 ± 0.35	34.11 ± 0.52
K-ScSPM	27.81 ± 0.37	34.93 ± 0.44
NDL	29.30 ± 0.29	36.80 ± 0.45
SNDL	**31.10** ± **0.35**	**38.25** ± **0.43**
Griffin [64]	28.30	34.10
ScSPM [13,33]	27.73 ± 0.51	34.02 ± 0.35

all datasets in the NDL method. In the SNDL method, the sizes of class-specific dictionary for each class were set as 60, 100, 20, and 20 for the Scene-15, UIUC-Sports, MIT67 Indoor, and Caltech-256 datasets, respectively.

5.3. Experimental results and analysis

In this subsection, we compare our NDL and SNDL methods with two baseline methods as:

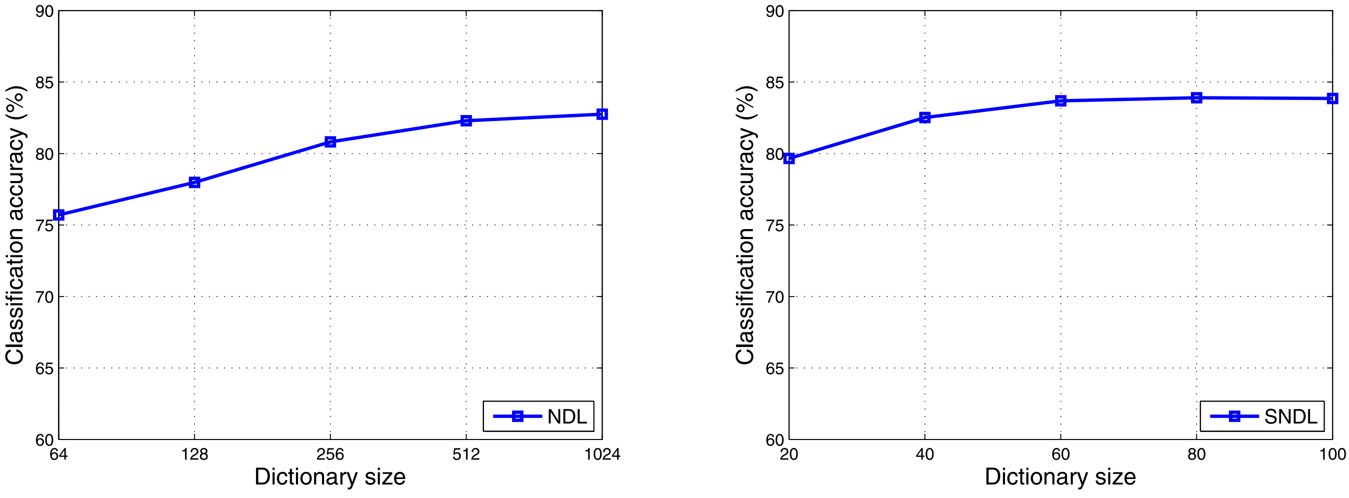
- Sparse coding with spatial pyramid matching (ScSPM) [13]: it uses the sparse coding in the original feature space under the general bag-of-visual-words (BoV) and spatial pyramid matching framework;
- Kernel based ScSPM (K-ScSPM) [25,33]: it employs the kernel trick to learn a dictionary in the transformed space using ScSPM approach. In our settings, we chose the RBF kernel to construct the kernel matric as $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|_2^2)$, where γ was set as the inverse of dimension of the vector $\mathbf{x}_i$.

For these baseline methods, we used the same dictionary size (K=1024) and the strategy of spatial pyramid representation as used in our approaches. Tables 1 and 3 summarize the classification results of both baselines and our proposed methods.

We can make the following observations from the results shown in Tables 1 and 3:

- Both the NDL and SNDL can obtain a better performance than that of ScSPM for image classification. The reason is that the ScSPM usually learns a dictionary and solves the sparse representation in the original space and it cannot well exploit the nonlinear structure of data, while our approaches can capture this nonlinearity via a feed-forward neural network so that the nonlinearity is better exploited.
- The NDL consistently performs better than K-ScSPM that uses kernel trick to implicitly encode the nonlinearity of data points, and the merit of the NDL is to explicitly seek hierarchical nonlinear transformations.
- The SNDL further improves the performance of the NDL, because the SNDL is a supervised dictionary learning method, and it can utilize the label information of training data to learn a discriminative dictionary for feature representation.

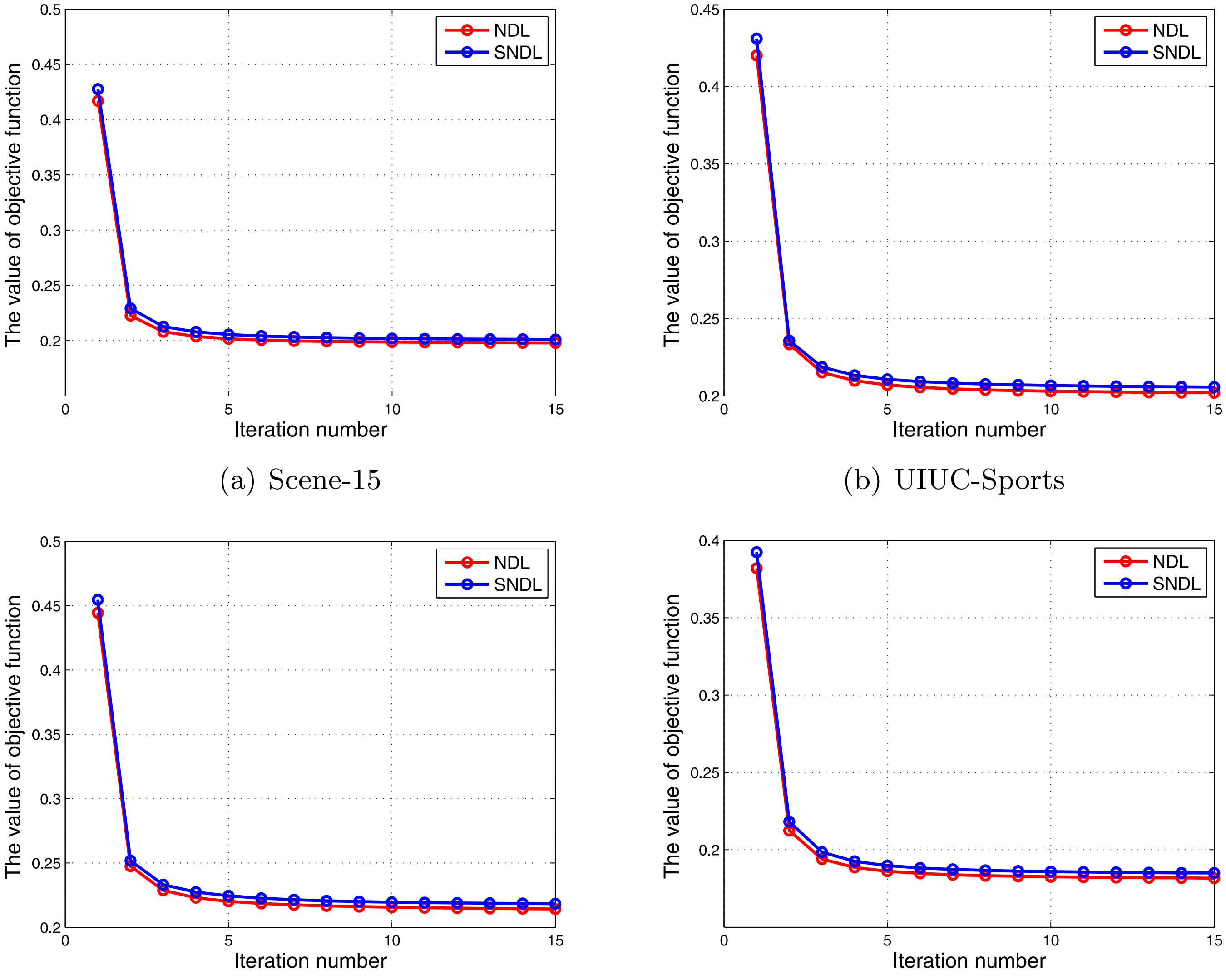
In addition, Tables 2 and 3 also show some representative results taken from the corresponding references, where most of these methods only use a single descriptor type as our methods. We can see that our methods are comparable to these methods, especially the DeepSC method which adopts the siamese network for hierarchical sparse coding.



(a) NDL

(b) SNDL

Fig. 4. The average classification accuracy (%) of the NDL and SNDL versus different dictionary sizes on the Scene-15 dataset.



(a) Scene-15

(b) UIUC-Sports

(c) MIT67 Indoor

(d) Caltech-256

Fig. 5. Convergence curves of the NDL and SNDL versus the number of iterations on the Scene-15, UIUC-Sports, MIT67 Indoor, and Caltech-256 datasets.

Table 4
The classification accuracy (%) of NDL and SNDL with different settings of layers on the Scene-15 dataset. The bold in Table denotes the best classification accuracy.

Layers	$M = 1$	$M = 2$	$M = 3$
NDL	82.12 ± 0.61	82.75 ± 0.53	82.76 ± 0.75
SNDL	**82.95 ± 0.77**	**83.69 ± 0.45**	**83.81 ± 0.56**

5.4. Parameter analysis

We also examine the performance of both the NDL and SNDL methods on the Scene-15 dataset versus several varying parameters.

5.4.1. Effect of the number of layers

To evaluate how the number of layers affects the classification performance of the NDL and SNDL methods, we used other two settings of the neural network: M=1: 128 → 128 → 128 and M=3: 128 → 128 → 128 → 128 → 128 → 128 → 128. Table 4 shows the classification accuracy of the NDL and SNDL by using three different settings of layers, we see that the performance of our methods is improved when the number of layers increases.

5.4.2. Effect of the size of the dictionary

Fig. 4 shows the average classification accuracy of the NDL and SNDL versus different dictionary sizes on the Scene-15 dataset. We see that the performance of the NDL can be improved by increasing the number of atoms, and it becomes stable when the dictionary size reaches 512. For the SNDL method, its classification accuracy also gradually increases when the size of class-specific dictionary varies from 20 to 60, and keeps relatively stable when the size of class-specific dictionary varies from 60 to 100.

5.4.3. Convergence analysis

Fig. 5 plots the objective function values of the NDL and SNDL methods versus different number of iterations on the Scene-15, UIUC-Sports, MIT67 Indoor, and Caltech-256 datasets. We see that our approaches can quickly converge after 5 iterations on these four datasets.

5.4.4. Parameter selection

We choose λ_1 in {0, 0.05, 0.1, 0.15, 0.2, 0.3, 0.5, 1}, λ_2 in {0, 0.1, 0.5, 1, 10} and λ_3 in {0, 0.1, 0.01, 0.001, 0.0001} on the Scene-15 dataset to examine the importance of each component. Experimental results show that the terms $\mathcal{J}_1$ and $\mathcal{J}_2$ are more important than $\mathcal{J}_3$. With the fixed $\lambda_1 = 0.15$ and $\lambda_2 = 1$, the mean accuracy of NDL varies in range [82.53, 82.75]; with the fixed $\lambda_1 = 0.15$ and $\lambda_3 = 0.001$, the accuracy varies in [79.61, 82.75], and small λ_2 leads to poor performance; and with the fixed $\lambda_2 = 1$ and $\lambda_3 = 0.001$, the accuracy varies in [81.85, 82.75]. For the learning rate β_1, we pick its initial value in {1, 0.1, 0.01}. We find that $\lambda_1 = 0.15$, $\lambda_2 = 1$, $\lambda_3 = 0.001$, and $\beta_1 = 0.1$ can obtain the acceptable results. Therefore we adopt this parameter setting for other datasets.

6. Conclusion

In this paper, we have proposed a nonlinear dictionary learning (NDL) method and applied it to image classification. The NDL employs a feed-forward neural network to seek hierarchical feature projection matrices and dictionary simultaneously, so that the nonlinear structure of samples can be well exploited. To better exploit the discriminative information, we have also extended NDL into supervised NDL (SNDL) by learning a class-specific dictionary with the label of training samples. Experimental results on four image classification datasets show the effectiveness of the proposed methods.

Acknowledgment

This work is partially supported by the Rapid-Rich Object Search (ROSE) Lab through the National Research Foundation, Singapore, under its Interactive Digital Media (IDM) Strategic Research Programme.

References

[1] M. Elad, M.A.T. Figueiredo, Y. Ma, On the role of sparse and redundant representations in image processing, Proc. IEEE 98 (6) (2010) 972–982, doi:10.1109/JPROC.2009.2037655.

[2] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T.S. Huang, S. Yan, Sparse representation for computer vision and pattern recognition, Proc. IEEE 98 (6) (2010) 1031–1044, doi:10.1109/JPROC.2010.2044470.

[3] M. Elad, M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, IEEE Trans. Image Process. 15 (12) (2006) 3736–3745, doi:10.1109/TIP.2006.881969.

[4] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227, doi:10.1109/TPAMI.2008.79.

[5] J. Lu, Y. Tan, G. Wang, Discriminative multimanifold analysis for face recognition from a single training sample per person, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 39–51, doi:10.1109/TPAMI.2012.70.

[6] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, J. Zhou, Cost-sensitive semi-supervised discriminant analysis for face recognition, IEEE Trans. Inf. Forensics Secur. 7 (3) (2012) 944–953, doi:10.1109/TIFS.2012.2188389.

[7] J. Lu, Y. Tan, G. Wang, G. Yang, Image-to-set face recognition using locality repulsion projections and sparse reconstruction-based similarity measure, IEEE Trans. Circuits Syst. Video Technol. 23 (6) (2013) 1070–1080, doi:10.1109/TCSVT.2013.2241353.

[8] J. Lu, Y. Tan, Gait-based human age estimation, IEEE Trans. Inf. Forensics Secur. 5 (4) (2010) 761–770, doi:10.1109/TIFS.2010.2069560.

[9] A. Moeini, H. Moeini, A.M. Safai, K. Faez, Regression facial attribute classification via simultaneous dictionary learning, Pattern Recognit. 62 (2017) 99–113, doi:10.1016/j.patcog.2016.08.031.

[10] J. Lu, G. Wang, P. Moulin, Human identity and gender recognition from gait sequences with arbitrary walking directions, IEEE Trans. Inf. Forensics Secur. 9 (1) (2014) 51–61, doi:10.1109/TIFS.2013.2291969.

[11] J. Lu, Y.-P. Tan, Ordinary preserving manifold analysis for human age and head pose estimation, IEEE Trans. Hum.-Mach. Syst. 43 (2) (2013) 249–258, doi:10.1109/TSMCC.2012.2192727.

[12] J. Mairal, F. Bach, J. Ponce, G. Sapiro, A. Zisserman, Discriminative learned dictionaries for local image analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8, doi:10.1109/CVPR.2008.4587652.

[13] J. Yang, K. Yu, Y. Gong, T.S. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1794–1801, doi:10.1109/CVPRW.2009.5206757.

[14] N. Akhtar, F. Shafait, A.S. Mian, Discriminative bayesian dictionary learning for classification, IEEE Trans. Pattern Anal. Mach. Intell. 38 (12) (2016) 2374–2388, doi:10.1109/TPAMI.2016.2527652.

[15] M. Swaminathan, P.K. Yadav, O. Piloto, T. Sjöblom, I. Cheong, A new distance measure for non-identical data with application to image classification, Pattern Recognit. 63 (2017) 384–396, doi:10.1016/j.patcog.2016.10.018.

[16] N. Wang, J. Wang, D. Yeung, Online robust non-negative dictionary learning for visual tracking, in: IEEE International Conference on Computer Vision, 2013, pp. 657–664, doi:10.1109/ICCV.2013.87.

[17] R. Rubinstein, A.M. Bruckstein, M. Elad, Dictionaries for sparse representation modeling, Proc. IEEE 98 (6) (2010) 1045–1057, doi:10.1109/JPROC.2010.2040551.

[18] S. Gao, I.W. Tsang, Y. Ma, Learning category-specific dictionary and shared dictionary for fine-grained image categorization, IEEE Trans. Image Process. 23 (2) (2014) 623–634, doi:10.1109/TIP.2013.2290593.

[19] Z. Jiang, Z. Lin, L.S. Davis, Label consistent K-SVD: learning a discriminative dictionary for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2651–2664, doi:10.1109/TPAMI.2013.88.

[20] H. Lee, A. Battle, R. Raina, A.Y. Ng, Efficient sparse coding algorithms, in: Advances in Neural Information Processing Systems, 2006, pp. 801–808.

[21] J. Mairal, F. Bach, J. Ponce, Task-driven dictionary learning, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2012) 791–804, doi:10.1109/TPAMI.2011.156.

[22] I. Ramírez, P. Sprechmann, G. Sapiro, Classification and clustering via dictionary learning with structured incoherence and shared features, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 3501–3508, doi:10.1109/CVPR.2010.5539964.

[23] M. Yang, L. Zhang, X. Feng, D. Zhang, Fisher discrimination dictionary learning for sparse representation, in: IEEE International Conference on Computer Vision, 2011, pp. 543–550, doi:10.1109/ICCV.2011.6126286.

[24] H. Wang, Y. Kawahara, C. Weng, J. Yuan, Representative selection with structured sparsity, Pattern Recognit. 63 (2017) 268–278, doi:10.1016/j.patcog.2016.10.014.

[25] H.V. Nguyen, V.M. Patel, N.M. Nasrabadi, R. Chellappa, Design of non-linear kernel dictionaries for object recognition, IEEE Trans. Image Process. 22 (12) (2013) 5123–5135, doi:10.1109/TIP.2013.2282078.

[26] M. Aharon, M. Elad, A. Bruckstein, K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation, IEEE Trans. Signal Process. 54 (11) (2006) 4311–4322, doi:10.1109/tsp.2006.881199.

[27] J. Mairal, F. Bach, J. Ponce, G. Sapiro, A. Zisserman, Supervised dictionary learning, in: Advances in Neural Information Processing Systems, 2008, pp. 1033–1040.

[28] L. Shen, S. Wang, G. Sun, S. Jiang, Q. Huang, Multi-level discriminative dictionary learning towards hierarchical visual categorization, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 383–390, doi:10.1109/CVPR.2013.56.

[29] M. Yang, D. Dai, L. Shen, L.V. Gool, Latent dictionary learning for sparse representation based classification, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 4138–4145, doi:10.1109/CVPR.2014.527.

[30] Q. Zhang, B. Li, Discriminative K-SVD for dictionary learning in face recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 2691–2698, doi:10.1109/CVPR.2010.5539989.

[31] Y. Boureau, F. Bach, Y. LeCun, J. Ponce, Learning mid-level features for recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 2559–2566, doi:10.1109/CVPR.2010.5539963.

[32] S. Bengio, F.C.N. Pereira, Y. Singer, D. Strelow, Group sparse coding, in: Advances in Neural Information Processing Systems, 2009, pp. 82–89.

[33] S. Gao, I.W. Tsang, L. Chia, Sparse representation with kernels, IEEE Trans. Image Process. 22 (2) (2013) 423–434, doi:10.1109/TIP.2012.2215620.

[34] G.B. Huang, H. Lee, E.G. Learned-Miller, Learning hierarchical representations for face verification with convolutional deep belief networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2518–2525, doi:10.1109/CVPR.2012.6247968.

[35] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Deepface: Closing the gap to human-level performance in face verification, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1701–1708, doi:10.1109/CVPR.2014.220.

[36] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, J. Zhou, Neighborhood repulsed metric learning for kinship verification, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 331–345, doi:10.1109/TPAMI.2013.134.

[37] J. Lu, V.E. Liong, X. Zhou, J. Zhou, Learning compact binary face descriptor for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (10) (2015a) 2041–2056, doi:10.1109/TPAMI.2015.2408359.

[38] J. Lu, V.E. Liong, J. Zhou, Cost-sensitive local binary feature learning for facial age estimation, IEEE Transactions on Image Processing 24 (12) (2015b) 5356–5368, doi:10.1109/TIP.2015.2481327.

[39] J. Lu, G. Wang, P. Moulin, Localized multifeature metric learning for image-set-based face recognition, IEEE Trans. Circuits Syst. Video Technol. 26 (3) (2016) 529–540, doi:10.1109/TCSVT.2015.2412831.

[40] J. Lu, V.E. Liong, G. Wang, P. Moulin, Joint feature learning for face recognition, IEEE Trans. Inf. Forensics Secur. 10 (7) (2015a) 1371–1383, doi:10.1109/TIFS.2015.2408431.

[41] J. Lu, G. Wang, W. Deng, K. Jia, Reconstruction-based metric learning for unconstrained face verification, IEEE Trans. Inf. Forensics Secur. 10 (1) (2015b) 79–89, doi:10.1109/TIFS.2014.2363792.

[42] J. Lu, Y.-P. Tan, Cost-sensitive subspace analysis and extensions for face recognition, IEEE Trans. Inf. Forensics Secur. 8 (3) (2013) 510–519, doi:10.1109/TIFS.2013.2243146.

[43] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, T. Darrell, DeCAF: A deep convolutional activation feature for generic visual recognition, in: International Conference on Machine Learning, 2014, pp. 647–655.

[44] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1106–1114.

[45] Q.V. Le, W.Y. Zou, S.Y. Yeung, A.Y. Ng, Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3361–3368, doi:10.1109/CVPR.2011.5995496.

[46] N. Wang, D. Yeung, Learning a deep compact image representation for visual tracking, in: Advances in Neural Information Processing Systems, 2013, pp. 809–817.

[47] Y. Bengio, A.C. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828, doi:10.1109/TPAMI.2013.50.

[48] G.E. Hinton, S. Osindero, Y.W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554, doi:10.1162/neco.2006.18.7.1527.

[49] R. Hadsell, S. Chopra, Y. LeCun, Dimensionality reduction by learning an invariant mapping, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 1735–1742, doi:10.1109/CVPR.2006.100.

[50] J. Hu, J. Lu, Y.-P. Tan, Discriminative deep metric learning for face verification in the wild, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1875–1882, doi:10.1109/CVPR.2014.242.

[51] J. Lu, G. Wang, W. Deng, P. Moulin, J. Zhou, Multi-manifold deep metric learning for image set classification, in: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1137–1145, doi:10.1109/CVPR.2015.7298717.

[52] J. Hu, J. Lu, Y.-P. Tan, Deep metric learning for visual tracking, IEEE Trans. Circuits Syst. Video Technol. 26 (11) (2016a) 2056–2068, doi:10.1109/TCSVT.2015.2477936.

[53] J. Hu, J. Lu, Y.-P. Tan, J. Zhou, Deep transfer metric learning, IEEE Trans. Image Process. 25 (12) (2016b) 5576–5588, doi:10.1109/TIP.2016.2612827.

[54] S. Gu, L. Zhang, W. Zuo, X. Feng, Projective dictionary pair learning for pattern classification, in: Advances in Neural Information Processing Systems, 2014, pp. 793–801.

[55] H. Wang, F. Nie, W. Cai, H. Huang, Semi-supervised robust dictionary learning via efficient l-norms minimization, in: IEEE International Conference on Computer Vision, 2013a, pp. 1145–1152, doi:10.1109/ICCV.2013.146.

[56] Z. Wang, J. Yang, N.M. Nasrabadi, T.S. Huang, A max-margin perspective on sparse representation-based classification, in: IEEE International Conference on Computer Vision, 2013b, pp. 1217–1224, doi:10.1109/ICCV.2013.154.

[57] Y. He, K. Kavukcuoglu, Y. Wang, A. Szlam, Y. Qi, Unsupervised feature learning by deep sparse coding, in: SIAM International Conference on Data Mining, 2014, pp. 902–910, doi:10.1137/1.9781611973440.103.

[58] K. Yu, Y. Lin, J.D. Lafferty, Learning image representations from the pixel level via hierarchical sparse coding, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 1713–1720, doi:10.1109/CVPR.2011.5995732.

[59] A. Argyriou, T. Evgeniou, M. Pontil, Convex multi-task feature learning, Mach. Learn. 73 (3) (2008) 243–272.

[60] H. Wang, F. Nie, H. Huang, Robust distance metric learning via simultaneous l1-norm minimization and maximization, in: International Conference on Machine Learning, 2014, pp. 1836–1844.

[61] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 2169–2178, doi:10.1109/CVPR.2006.68.

[62] L. Li, F. Li, What, where and who? classifying events by scene and object recognition, in: IEEE International Conference on Computer Vision, 2007, pp. 1–8, doi:10.1109/ICCV.2007.4408872.

[63] A. Quattoni, A. Torralba, Recognizing indoor scenes, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 413–420, doi:10.1109/CVPRW.2009.5206537.

[64] G. Griffin, A. Holub, P. Perona, Caltech-256 Object Category Dataset, Technical Report 7694, California Institute of Technology, 2007.

[65] L. Li, H. Su, E.P. Xing, F. Li, Object bank: a high-level image representation for scene classification & semantic feature sparsification, in: Advances in Neural Information Processing Systems, 2010, pp. 1378–1386.

Junlin Hu received the B.Eng. degree from the Xi'an University of Technology, Xi'an, China, in 2008, and the M.Eng. degree from Beijing Normal University, Beijing, China, in 2012. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His current research interests include computer vision, pattern recognition, and biometrics.

Yap-Peng Tan received the B.S. degree from National Taiwan University, Taipei, Taiwan, in 1993, and the M.A. and Ph.D. degrees from Princeton University, Princeton, NJ, USA, in 1995 and 1997, respectively, all in electrical engineering. He was with Intel Corporation, Chandler, AZ, USA, and Sharp Laboratories of America, Camas, WA, USA, from 1997 to 1999. In 1999, he joined Nanyang Technological University, Singapore, where he is currently an Associate Professor and the Associate Chair (Academic) of the School of Electrical and Electronic Engineering. He is the principal inventor or co-inventor on 15 U.S. patents in the areas of image and video processing. His current research interests include image and video processing, content-based multimedia analysis, computer vision, and pattern recognition. Dr. Tan was a recipient of an IBM Graduate Fellowship from the IBM T. J. Watson Research Center, Yorktown Heights, NY, USA, from 1995 to 1997. He served as the Chair of the Visual Signal Processing and Communications Technical Committee of the IEEE Circuits and Systems Society, a member of the Multimedia Signal Processing Technical Committee of the IEEE Signal Processing Society, and a Voting Member of the ICME Steering Committee. He is an Editorial Board Member of the IEEE Transactions on Multimedia, the *EURASIP Journal on Advances in Signal Processing*, and the *EURASIP Journal on Image and Video Processing*, and an Associate Editor of the *Journal of Signal Processing Systems*. He was the General Co-Chair of the 2010 IEEE International Conference on Multimedia and Expo and is the Co-Chair of the 2015 IEEE Conference on Visual Communications and Image Processing. He was a Guest Editor of special issues of several journals, including the IEEE Transactions on Multimedia.