Deep Robust Encoder Through Locality Preserving Low-Rank Dictionary

Zhengming Ding[1(✉)], Ming Shao[1], and Yun Fu[1,2]

[1] Department of Electrical and Computer Engineering, Northeastern University, Boston, USA
{allanding,mingshao}@ece.neu.edu
[2] College of Computer and Information Science, Northeastern University, Boston, USA
yunfu@ece.neu.edu

Abstract. Deep learning has attracted increasing attentions recently due to its appealing performance in various tasks. As a principal way of deep feature learning, deep auto-encoder has been widely discussed in such problems as dimensionality reduction and model pre-training. Conventional auto-encoder and its variants usually involve additive noises (e.g., Gaussian, masking) for training data to learn robust features, which, however, did not consider the already corrupted data. In this paper, we propose a novel Deep Robust Encoder (DRE) through locality preserving low-rank dictionary to extract robust and discriminative features from corrupted data, where a low-rank dictionary and a regularized deep auto-encoder are jointly optimized. First, we propose a novel loss function in the output layer with a learned low-rank clean dictionary and corresponding weights with locality information, which ensures that the reconstruction is noise free. Second, discriminant graph regularizers that preserve the local geometric structure for the data are developed to guide the deep feature learning in each encoding layer. Experimental results on several benchmarks including object and face images verify the effectiveness of our algorithm by comparing with the state-of-the-art approaches.

Keywords: Auto-encoder · Low-rank dictionary · Graph regularizer

1 Introduction

In the recent years, deep learning has attracted considerable interests in computer vision field, as it has achieved promising performance in various tasks, e.g., image classification [1], object detection [2] and face recognition [3]. Generally, deep structure learning tends to extract hierarchical feature representations directly from raw data. Recent representative research works include: deep convolutional neural networks [4], deep neural networks [5], deep auto-encoder [6], and deeply-supervised nets [7].

B. Leibe et al. (Eds.): ECCV 2016, Part VI, LNCS 9910, pp. 567–582, 2016.
DOI: 10.1007/978-3-319-46466-4_34

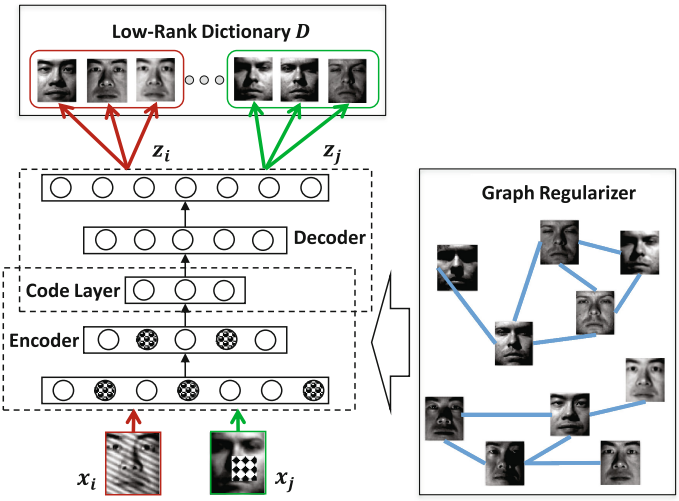

Fig. 1. Illustration of our proposed algorithm. Corrupted data x_i, x_j are the inputs of the deep AE. After encoding and decoding process, the reconstructed x_i, x_j are encouraged to be close to Dz_i, Dz_j on the top, where D is the learned clean low-rank dictionary and z_i, z_j are corresponding coefficients. In addition, graph regularizers are added to the encoder layers to pass on the locality information.

Among different deep structures, auto-encoder (AE) [8] has been treated as robust feature extractors or pre-training scheme in various tasks [9–14]. Conventional AE was proposed to encourage similar or identical input-output pairs where the reconstruction loss is minimized after decoding [8]. Follow-up work with various additive noises in the input layer is able to progressively purify the data, which fulfills the purpose "denoising" against unknown corruptions in the testing data [15]. These works as well as the most recent AE variants, e.g., multi-view AE [13] and bi-shift AE [11], all assume the training data are clean, but can be intentionally corrupted. In fact, real-world data subject to corruptions such as changing illuminations, pose variations, or self-corruption do not meet the assumption above. Therefore, learning deep features from real-world corrupted data instead of intentionally corrupted data with additive noises becomes critical to build robust feature extractor that is generalized well to corrupted testing data. To the best of our knowledge, such AE based deep learning scheme has not been discussed before.

Recently, low-rank matrix constraint has been proposed to learn robust features from corrupted data. Specifically, when data are lying in a single subspace, robust PCA (RPCA) [16] could well recover the corrupted data by seeking a low-rank basis. While low-rank representation (LRR) [17] is designed to recover corrupted data and rule out noises in case of multiple subspaces. Due to these technical merits, low-rank modeling has already been successfully used in different scenarios, e.g., multi-view learning [18], transfer learning [19–21], and

dictionary learning [22]. However, fewer works link the low-rank modeling to deep learning framework for robust feature learning.

Inspired by the above facts, we develop a novel algorithm named as Deep Robust Encoder (DRE) with locality preserving low-rank dictionary. The core idea is to jointly optimize deep AE and a clean low-rank dictionary, which can rule out noises and extract robust deep features in a unified framework (Fig. 1). To sum up, our contributions are three folds as follows:

- A low-rank dictionary and deep AE are jointly optimized based on the corrupted data, which can progressively denoise the already corrupted features in the hidden layers so that robust deep AE could be achieved for corrupted testing data.
- The newly designed loss function, which is based on the clean low-rank dictionary and preserved locality information in the output layer, penalizes the corruptions or distortions, meanwhile ensures that the reconstruction is noise free.
- Graph regularizers are developed to guide feature learning in each encoding layer to preserve more geometric structures within the data, in either unsupervised or supervised fashions.

The remaining sections of this paper are organized as follows. In Sect. 2, we present a brief discussion of the related works. Then we propose our novel deep robust encoder in Sect. 3, as well as the solution. Experimental evaluations are reported in Sect. 4, followed by the conclusion in Sect. 5.

2 Related Work

In this section, we mainly discuss the recent related works and highlight the differences between their approaches and ours.

Auto-encoder (AE) has attracted lots of research interests in computer vision fields. It was recently proposed as an efficient scheme for deep structures pretraining and dimensionality reduction [5,8]. Denoising auto-encoder (DAE) generated a robust feature extractor by incorporating artificially random noise to the input data, and then minimized the square loss between reconstructed output and original clean data [15]. Most recently, appealing AE variants have been proposed to handle different learning tasks, e.g., transfer learning [11], domain generalization [12] and multi-view learning [13]. Generally, these variants aim to adapt the knowledge from one domain/view to another by tuning the input or the target data. Different from them, we consider that the real-world data already have been corrupted somehow and we develop an active deep denoising framework to handle the existing corruptions in the training data, which can then be well generalized to the unseen corrupted testing data. However, to the best knowledge, little has been discussed with regard to AE.

Low-rank modeling has demonstrated with appealing performance on robust feature extraction against noisy data. Recently, Robust PCA (RPCA) [16] has been proposed to rule out noises for data lying in a single subspace. Moreover,

low-rank representation (LRR) [17] is presented recently to handle real-world noisy data lying in multiple subspaces. It can identify the global subspace structure as well as corruptions. Besides, low-rank modeling has also been adopted in different learning tasks, e.g., generic feature extraction [18], visual domain adaptation [21], robust transfer learning [19], and dictionary learning [22]. In this paper, we also involve the low-rank constraint on the dictionary learning to build a clean and compact basis. Differently, we exploit the low-rank dictionary to reconstruct the outputs of the deep AE with corrupted inputs, instead of the original data [22]. In this way, we could build an active deep denoising framework to generate more robust features from corrupted data. Furthermore, locality-preserving reconstruction helps maintain the geometric structure of the data, which has not been discussed with low-rank dictionary in deep learning before.

3 The Proposed Algorithm

In this section, we first introduce our motivation, and then propose our deep robust encoder through locality preserving low-rank dictionary. Finally, we present an efficient solution to the proposed framework.

3.1 Motivation

Intentional corruptions, e.g., random noises are added artificially while real-world ones are from data itself, e.g., varied lightings or occlusion. Most existing AE and its variants, e.g., DAE, take advantage of different additive noises on the clean data to improve the robustness of deep models. During the deep encoding/decoding process, the perturbed input data are gradually recovered. In this way, the learned deep model is able to tolerate certain corruptions simulated by the additive noises.

However, this raises two problems. First, the robustness of the system completely relies on the formulations of the noises. The richer the noisy patterns are, the better the performance will be. This inevitably increases the computational burden. In the worst case, the learned deep structure may not be well generalized to the unseen testing data. Second, real-world data usually suffer from contaminations of varied sources, and building robust feature extractors to rule out existing noises is more reasonable. In addition, recent advances in low-rank matrix modeling cast a light on denoising for data that are already corrupted. Based on these observations, we propose to jointly learn a deep AE framework and a clean low-rank dictionary to actively mitigate the noises or corruptions within the data (Fig. 2).

3.2 Locality Preserving Low-Rank Dictionary Learning

Suppose training data $X \in \mathbb{R}^{d \times n}$ has n samples and $x_i \in \mathbb{R}^d$ represents the i-th sample. For AE with single hidden layer [5,8], it is usually consisted of two parts,

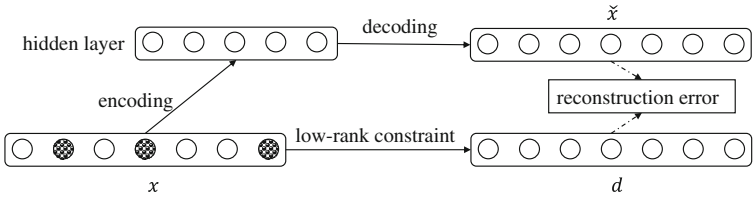

Fig. 2. The AE architecture with low-rank dictionary. A corrupted sample x is correlated to a low-rank clean version d. The AE then maps it to hidden layer (via encoder layer) and attempts to reconstruct x via decoder layer, generating reconstruction $\check{x}$. Finally, reconstruction error can be measured by different loss functions.

encoder and decoder. The encoder, denoted as f_1, attempts to map the input x_i into hidden representations, while the decoder, denoted as f_2, tries to map the hidden representation back to the input x_i. A typical cost function with square loss for AE can be formulated as:

$$\min_{W_1,b_1,W_2,b_2} \sum_{i=1}^{n} \left\| x_i - f_2(f_1(x_i)) \right\|_2^2, \tag{1}$$

where $\{W_1 \in \mathbb{R}^{r\times d}, b_1 \in \mathbb{R}^r\}, \{W_2 \in \mathbb{R}^{d\times r}, b_2 \in \mathbb{R}^d\}$ are the parameters for encoding and decoding, respectively. Specifically, we have $f_1(x_i) = \varphi(W_1 x_i + b_1)$ and $f_2(f_1(x_i)) = \varphi(W_2 f_1(x_i) + b_2)$, where $\varphi(\cdot)$ is an element-wise "activation function", which is usually nonlinear, such as sigmoid function or tanh function. DAE manually involves artificial noise into the input training data so that it aims to train a denoising auto-encoder to remove the random noise.

In reality, however, x_i is usually corrupted already due to environmental factors or noises from the collecting devices. Intuitively, we need to build a network by detecting and removing noise from the corrupted data so that it could better generalize to corrupted testing data. To this end, we propose our robust auto-encoder with low-rank dictionary learning:

$$\min_{W_1,b_1,W_2,b_2,D} \sum_{i=1}^{n} \| d_i - f_2(f_1(x_i)) \|_2^2 + \lambda \mathrm{rank}(D), \tag{2}$$

where $d_i \in \mathbb{R}^d$ is the i-th column of low-rank $D \in \mathbb{R}^{d\times n}$ and λ is the tradeoff parameter. $\mathrm{rank}(\cdot)$ means the rank operator of a matrix, which encourages to build a clean and compact basis. Generally, the convex surrogate of rank problem, i.e., nuclear norm $\|\cdot\|_*$ will be employed to solve the rank minimization problem [16].

However, similar to the conventional AE and its variants, the point-to-point reconstruction scheme in Eq. (2) only considers one-to-one mapping, which may overfit the data and skip the structure knowledge within the data. To that end, we propose a novel locality preserving low-rank dictionary learning by introducing a new coefficient vector z_i to maintain the locality of each sample x_i throughout the network:

$$\min_{W_1,b_1,W_2,b_2,D} \sum_{i=1}^{n} \|Dz_i - f_2(f_1(x_i))\|_2^2 + \lambda\|D\|_*, \tag{3}$$

where $z_i \in \mathbb{R}^n$ is the coefficient vector for sample x_i w.r.t. dictionary D. There are different strategies to obtain the coefficient vector z_i, in either unsupervised or supervised fashion, depending on the availability of label information. Specifically, the j-th element in z_i is defined as:

$$z_{ij} = \begin{cases} 1, & \text{if } i = j, \\ \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right), & \text{if } x_i \in \mathcal{N}_{k_1}(x_j), \\ 0, & \text{otherwise}, \end{cases} \tag{4}$$

where $x_i \in \mathcal{N}_{k_1}(x_j)$ means x_i is within the k_1 nearest neighbors of x_j. Specifically, we could define the locality-preserving coefficients z_i in two fashions. For unsupervised case, the k_1 nearest neighbors are searched from the whole data, while for supervised case, the k_1 nearest neighbors are searched from the data within the same class to x_i. Actually, we could easily extend semi-supervised scenario. Note σ is a bandwidth for Gaussian kernel (we set $\sigma = 5$ in this paper).

To sum up, our regularized deep auto-encoder transform the original AE's point-to-point reconstruction strategy to our point-to-set reconstruction so that we could preserve more discriminative information. To further guide the locality preserving dictionary learning in the output layer, we propose to couple the discriminant graph regularizers with hidden feature learning during the optimization:

$$\begin{aligned} \min_{W_1,b_1,W_2,b_2,D} & \sum_{i=1}^{n} \|Dz_i - f_2(f_1(x_i))\|_2^2 + \lambda\|D\|_* \\ & + \alpha \sum_{j=1}^{n} \sum_{k=1}^{n} s_{jk}(f_1(x_j) - f_1(x_k))^2, \end{aligned} \tag{5}$$

where s_{jk} is the similarity between x_j and x_k. α is the balance parameter.

Specifically, s_{jk} can be calculated in unsupervised and supervised fashions as well:

$$s_{jk} = \begin{cases} \exp\left(-\frac{\|x_j - x_k\|^2}{2\sigma^2}\right), & \text{if } x_j \in \mathcal{N}_{k_2}(x_k), \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

where $x_j \in \mathcal{N}_{k_2}(x_k)$ means x_j is within the k_2 nearest neighbors of x_k. In the same way as z_i, the k_2 nearest neighbors are selected from the whole dataset for unsupervised case, while the k_2 nearest neighbors are selected from the data within the same class to x_j for supervised case.

3.3 Deep Architecture

Considering the learning objective in Eq. (5) as a basic building block, we can train a more discriminant deep model. Existing popular training schemes for deep auto-encoder includes Stacked Auto-Encoder (SAE) [15] and Deep Auto-Encoder [6]. However, as our learning objective/building block is different from theirs, we have a different training scheme for the deep structure.

Assume we have L encoding layers and L decoding layers in our deep structure which minimizes the following loss:

$$\begin{aligned}\min_{W_l,b_l,D} &\sum_{i=1}^{n} \|Dz_i - \bar{x}_i\|_2^2 + \lambda\|D\|_* \\ +\alpha &\sum_{l=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n} s_{jk}(f_l(x_j) - f_l(x_k))^2,\end{aligned} \tag{7}$$

where $\bar{x}_i$ is the output with a series of encoding and decoding from the input x_i. $\{W_l, b_l\}, (1 \le l \le L)$ are the encoding parameters while $\{W_l, b_l\}, (L{+}1 \le l \le 2L)$ are the decoding parameters. The third term sums up the graph regularizers from each encoding layer to guide the locality preserving low-rank dictionary learning in the output layer.

3.4 Optimization

Equation (7) is difficult to address because of the non-convexity and non-linearity of the building block formulated in Eq. (5). To this end, we develop an alternating solution to iteratively update the encoding & decoding functions $f_l(1 \le l \le 2L)$ and dictionary D. First we list the low-rank dictionary learning, then provide the regularized deep auto-encoder optimization.

Low-Rank Dictionary Learning. When $f_l(1 \le l \le 2L)$ are fixed, the objective function in Eq. (7) degenerates to a conventional low-rank recovery problem, which can be solved by augmented Lagrange multiplier algorithm [23]. To that end, we first involve a relaxing variable J, and write down its equivalent formulation as:

$$\min_{D,J} \|\bar{X} - DZ\|_{\mathrm{F}}^2 + \lambda\|J\|_*, \quad \text{s.t.} \quad D = J,$$

where $\bar{X} = [\bar{x}_1, \cdots, \bar{x}_n]$ and $Z = [z_1, \cdots, z_n]$. $\|\cdot\|_{\mathrm{F}}^2$ is Frobenius norm of a matrix. Then we derive the corresponding augmented Lagrangian function w.r.t. D, J:

$$\|\bar{X} - DZ\|_{\mathrm{F}}^2 + \lambda\|J\|_* + \langle R, D - J\rangle + \frac{\mu}{2}\|D - J\|_{\mathrm{F}}^2,$$

where R is the Lagrange multiplier and $\mu > 0$ is the penalty parameter. $\langle , \rangle$ is the matrix inner product operator. Specifically, we have the following updating rules for D, J one variable at time t:

$$J_{t+1} = \arg\min_{J} \frac{\lambda}{\mu_t}\|J\|_* + \frac{1}{2}\|J - D_t - \frac{R_t}{\mu_t}\|_{\mathrm{F}}^2, \tag{8}$$

which can be effectively addressed by the singular value thresholding (SVT) operator [24].

$$\begin{aligned} D_{t+1} &= \arg\min_{D} \|\bar{X} - DZ\|_{\mathrm{F}}^2 + \langle R_t, D - J_{t+1}\rangle + \tfrac{\mu_t}{2}\|D - J_{t+1}\|_{\mathrm{F}}^2 \\ &= (2\bar{X}Z^\top + \mu_t J_{t+1} - R_t)(2ZZ^\top + \mu_t \mathrm{I}_n)^{-1}, \end{aligned} \tag{9}$$

where $\mathrm{I}_n \in \mathbb{R}^{n\times n}$ is an identical matrix.

Deep Robust Encoder Learning. When D is fixed, the objective function in Eq. (7) can be reformulated to minimize the following objective function:

$$\mathcal{L} = \sum_{i=1}^{n} \|\bar{x}_i - \bar{d}_i\|_2^2 + \alpha \sum_{l=1}^{L} \sum_{j=1}^{n} \sum_{k=1}^{n} s_{jk}(f_l(x_j) - f_l(x_k))^2,$$

where $\bar{d}_i = Dz_i$. Since the loss function (Eq. (3.4)) is smooth and twice-differentiable, we can still adopt L-BFGS optimizer [25] to deal with this unconstrained problem, whose updating rules at time t are shown as follows:

$$\begin{cases} W_{l,t+1} = W_{l,t} - \eta_t H_{l,t} \dfrac{\partial \mathcal{L}}{\partial W_l}|_{W_{l,t}}, \\ b_{l,t+1} = b_{l,t} - \eta_t G_{l,t} \dfrac{\partial \mathcal{L}}{\partial b_l}|_{b_{l,t}}, \end{cases} \tag{10}$$

in which η_t denotes the learning rate, $H_{l,t}$ and $G_{l,t}$ are the approximations for the inverse Hessian matrices of $\mathcal{L}$ w.r.t. to W_l and b_l, respectively. The detailed formulations and discussions of η_t, $H_{l,t}$ and $G_{l,t}$ are trivial, which can be referred to [25]. In this section, we mainly focus on the derivatives of $\mathcal{L}$ w.r.t. to W_l and b_l.

For the **decoding layers** ($L+1 \leq l \leq 2L$), we have:

$$\frac{\partial \mathcal{L}}{\partial W_l} = \textstyle\sum_{i=1}^{n} \mathcal{F}_{i,l} \mathbf{f}_{i,l-1}^{\top}, \qquad \frac{\partial \mathcal{L}}{\partial b_l} = \textstyle\sum_{i=1}^{n} \mathcal{F}_{i,l},$$

where $\mathbf{f}_{i,l-1} = f_{l-1}(x_i)$ is the l-1$^{\text{th}}$-layer hidden layer feature and the updating equations are computed as follows:

$$\begin{aligned} \mathcal{F}_{i,2L} &= 2(\bar{x}_i - \bar{d}_i) \odot \varphi'(\mathbf{u}_{i,2L}), \\ \mathcal{F}_{i,l} &= (W_{l+1}^{\top} \mathcal{F}_{i,l+1}) \odot \varphi'(\mathbf{u}_{i,l}). \end{aligned}$$

Here the operator $\odot$ denotes the element-wise multiplication, and $\mathbf{u}_{i,l}$ is computed by $\mathbf{u}_{i,l} = W_l \mathbf{f}_{i,l-1} + b_l$.

For the **encoding layers** ($1 \leq l \leq L$), we have:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial W_l} &= \sum_{i=1}^{n} \mathcal{F}_{i,l} \mathbf{f}_{i,l-1}^{\top} + \\ &\qquad 2\alpha \sum_{p=l}^{L} \sum_{j=1}^{n} \sum_{k=1}^{n} s_{jk}(\mathcal{G}_{jk,p} \mathbf{f}_{j,p-1}^{\top} + \mathcal{G}_{kj,p} \mathbf{f}_{k,p-1}^{\top}), \\ \frac{\partial \mathcal{L}}{\partial b_l} &= \sum_{i=1}^{n} \mathcal{F}_{i,l} + 2\alpha \sum_{p=l}^{L} \sum_{j=1}^{n} \sum_{k=1}^{n} s_{jk}(\mathcal{G}_{jk,p} + \mathcal{G}_{kj,p}), \end{aligned}$$

in which $\mathcal{G}_{jk,l}$ and $\mathcal{G}_{kj,l}$ are calculated as follows:

$$\begin{aligned} \mathcal{G}_{jk,L} &= (\mathbf{f}_{j,l} - \mathbf{f}_{k,l}) \odot \varphi'(\mathbf{u}_{j,L}), \\ \mathcal{G}_{kj,L} &= (\mathbf{f}_{k,l} - \mathbf{f}_{j,l}) \odot \varphi'(\mathbf{u}_{k,L}), \\ \mathcal{G}_{jk,l} &= (W_{l+1}^{\top} \mathcal{G}_{jk,l+1}) \odot \varphi'(\mathbf{u}_{j,l}), \\ \mathcal{G}_{kj,l} &= (W_{l+1}^{\top} \mathcal{G}_{kj,l+1}) \odot \varphi'(\mathbf{u}_{k,l}). \end{aligned}$$

To that end, we can optimize low-rank dictionary and deep auto-encoder iteratively until convergence. The entire procedure of two sub-problems is listed in **Algorithm 1**. Before the alternative updating, the network parameters $f_l (1 \leq l \leq 2L)$ are initialized through deep auto-encoder with the input and the target as X [6], whilst D is directly set as original data X for initialization.

Algorithm 1. Solution to Problem (7)

Input: $\{X, y\}$, α, λ, $\eta_0 = 0.2, \varepsilon = 10^{-6}, t = 0$, $\mu_0 = 10^{-6}, \rho = 1.3, \mu_{\max} = 10^6$, and $t_{\max} = 10^3$.

while not converged **or** $t < t_{\max}$ **do**

Step 1. Update low-rank dictionary via (8),(9);

Step 2. Update the deep auto-encoder:

for $l = 2L, \cdots, 1$ **do**

Compute derivatives $\frac{\partial \mathcal{L}}{\partial W_l}, \frac{\partial \mathcal{L}}{\partial b_l}$;

end

for $l = 1, \cdots, 2L$ **do**

Update W_l, b_l using (10);

end

Step 3. Update parameters:

$R_{t+1} = R_t + \mu_t(D_{t+1} - J_{t+1})$; $\eta_{t+1} = 0.95 \times \eta_t$;

$\mu_{t+1} = \min(\mu_{\max}, \rho\mu_t)$; $t = t + 1$.

Step 4. Check convergence:

$|\mathcal{L}_{t+1} - \mathcal{L}_t| < \varepsilon$, $\|D_{t+1} - J_{t+1}\|_\infty < \varepsilon$.

end

Output: $\{W_l, b_l, D, J\}$.

4 Experiments

In this section, we conduct experiments to systematically evaluate our algorithm. First, we present the details of datasets and experimental settings. Then we do self-evaluation on our algorithm and present the comparison results with several state-of-the-art algorithms. Finally, we further testify several properties of the proposed algorithm, e.g., impacts of layer size, parameter analysis.

4.1 Datasets and Experimental Settings

COIL dataset[1] includes 72 views from 100 objects with different illumination conditions (Fig. 3). Each object is captured in equally spaced views, i.e., 5 degrees. In our experiments, we adopt the gray-scale images and resize them to 32×32. We randomly select ten images per object to build the training set, and the rest images as the testing set. We repeat the random selection process 20 times, and report the average performance. In addition, we perform scalability evaluations by gradually involving more categories from 20 to 100. Furthermore,

[1] http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php.

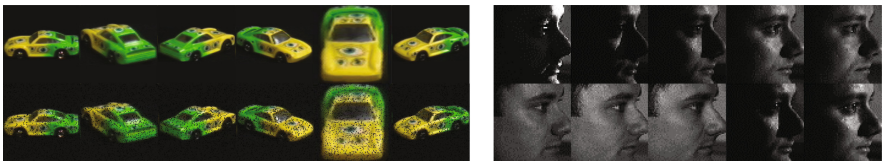

Fig. 3. Samples of two datasets: COIL-100 (left) and CMU-PIE (right). We show original images and 10 % corrupted ones for COIL-100. For CMU-PIE, the original faces already show large variance with one subject.

Table 1. Recognition results (%) of 4 approaches on different setting of three datasets.

	COIL-100c	PIE-1	PIE-2	PIE-1c	PIE-2c	ALOI-c
AE	74.56 ± 0.38	83.58 ± 0.11	82.79 ± 0.13	74.95 ± 0.14	73.89 ± 0.12	80.98 ± 0.98
LAE	78.32 ± 0.46	85.87 ± 0.16	85.08 ± 0.14	77.82 ± 0.12	76.14 ± 1.45	82.84 ± 1.26
L^2AE-u	79.84 ± 0.64	86.98 ± 0.09	86.45 ± 0.11	79.23 ± 0.11	79.02 ± 0.12	83.42 ± 0.87
L^2AE-s	82.42 ± 0.72	87.67 ± 0.10	87.54 ± 0.12	80.14 ± 0.10	79.96 ± 0.11	86.27 ± 0.75

we also evaluate the robustness of different approaches to noise by adding 10 % random corruption to the original images.

CMU-PIE Face dataset[2] contains 68 subjects under different poses subject to large appearance differences (Fig. 3). In addition, for each pose, there are 21 various illumination conditions. We use face images from 8 different poses to construct various evaluation sets. The sizes of them vary from 2 to 5. Basically, we randomly select 15 images per pose per subject to build the training set while the left as the testing set. The face images are cropped and resized to 64×64, and the raw features are used as the inputs.

ALOI dataset[3] consists of 1000 object categories captured from different viewing angles. Specifically, each object has 72 equally spaced views. In this experiments, we select the first 300 objects by following the setting in [26], where the images are transformed to gray-scale and resized to 36×48. Furthermore, 10 % pixel corruption is added to testify the robustness of different methods.

Note that previous algorithms, e.g., DAE [15], adopted the "corrupted" data with random noise as the input for training while using the "original" data for testing. However, we assume the data are "already corrupted" and we manage to detect and remove the noise. Thus, we adopt the "same" types of training and testing data without intentional corruptions. Notably, to challenge all comparisons, we introduce additional noises to the datasets that have already been corrupted by poor lighting or arbitrary views. Such practice can be found in previous work [22,26].

[2] http://vasc.ri.cmu.edu/idb/html/face/.
[3] http://aloi.science.uva.nl/.

4.2 Self-evaluation

In this section, we mainly testify if our low-rank dictionary D and locality preserving term $Z = [z_1, \cdots, z_n]$ would facilitate our robust feature learning. Specifically, we define the deep version of Eq.(2) as LAE (Auto-encoder with low-rank dictionary) and deep version of Eq. (3) as L^{2}AE (Auto-encoder with locality preserving low-rank dictionary). For L^{2}AE, we have two ways to learn Z, that is, we set $k_1 = k_2 = 5$ for all cases in unsupervised fashion (L^{2}AE-u), while we set k_1, k_2 as the size of each class for supervised fashion (L^{2}AE-s). A four-layer scheme is applied for all the comparisons for simplicity. We adopt corrupted COIL-100 and ALOI, while both original and corrupted images of CMU-PIE to testify these algorithms with the baseline, conventional AE [8]. The comparison results are shown in Table 1, where COIL-100c means the 10 % corrupted COIL using 100 objects, PIE-1 and PIE-2 denote the two views cases $\{C02, C14\}, \{C02, C27\}$ with its 10 % corrupted versions PIE-1c and PIE-2c, respectively. ALOI-c represents the 10 % corrupted data.

From the results, we could observe that LAE outperforms the conventional AE, that means jointly learning the low-rank dictionary could boost the deep feature learning of auto-encoder. Furthermore, we witness that our robust AEs with locality preserving low-rank dictionary could achieve better performance than LAE and AE for both unsupervised and supervised settings. That is, locality preserving property could generate more discriminative features for classification.

4.3 Comparison Experiments

We mainly compare with (1) traditional feature extract methods: PCA [27], LDA [28]; (2) low-rank based algorithms: RPCA+LDA [16], LatLRR [29], DLRD [22], LRCS [18], SRRS [26]. Specifically, PCA, LDA, RPCA+LDA, LRCS and SRRS belong to dimensionality reduction algorithms so that we search the optimal dimensionality for each to report the performance. Besides, to further evaluate the effectiveness of our algorithm, DAE [15] is adopted as the baseline. For our algorithm, we have two modes, i.e., unsupervised mode (Ours-I), and supervised mode (Ours-II). Specifically, we set parameters $\alpha = 10^2, \lambda = 10^{-2}$. For DAE and our two modes, we apply a four-layer deep structure. For Ours-I, we set $k_1 = k_2 = 5$ for all cases, while for Ours-II, we set k_1, k_2 as the size of each class. We apply the nearest neighbor classifier (NNC) for all algorithms except DLRD and show experimental results in Tables 2 and 3 and Fig. 4(a).

From Tables 2 and 3 and Fig. 4(a), we could observe our proposed algorithm in two modes outperforms others in most cases, especially for the corruption cases. In the corruption cases, our method has a significant improvement over others on two datasets (about 7 % improvement on corrupted COIL dataset). All the algorithms suffer from additional noises; however, ours can still achieve appealing performance (only 1–2 % performance degradation), which demonstrates the superiority of our method against noises in feature learning.

Table 2. Recognition results (%) of 9 algorithms on COIL-100 in different evaluation sizes, from 20 to 100 objects, where C1 to C5 denote 20 objects to 100 objects, respectively. Red color denotes the best recognition rates. Blue color denotes the second best.

	Original images								
	PCA	LDA	RPCA+LDA	DLRD	LatLRR	SRRS	DAE	Ours-I	Ours-II
C1	86.42 ± 1.11	81.83 ± 2.03	83.26 ± 1.52	89.58 ± 1.04	88.98 ± 0.85	92.03 ± 1.21	87.81 ± 1.43	90.65 ± 1.34	92.63 ± 0.95
C2	83.75 ± 1.12	77.08 ± 1.36	78.39 ± 1.15	85.18 ± 1.10	88.45 ± 0.64	92.51 ± 0.65	84.77 ± 1.25	90.34 ± 1.12	92.21 ± 0.63
C3	81.01 ± 0.92	66.96 ± 1.52	68.93 ± 0.86	82.60 ± 1.06	86.36 ± 0.52	90.82 ± 0.43	80.85 ± 0.65	87.69 ± 0.82	89.83 ± 0.52
C4	80.53 ± 0.78	59.34 ± 1.22	60.73 ± 0.68	81.10 ± 0.58	84.67 ± 0.79	88.75 ± 0.71	79.75 ± 0.61	85.15 ± 0.60	88.96 ± 0.91
C5	82.75 ± 0.59	52.29 ± 0.30	56.44 ± 0.73	79.92 ± 0.93	82.64 ± 0.60	85.12 ± 0.33	78.99 ± 0.48	84.21 ± 0.69	86.02 ± 0.61
	Corrupted images with 10 % random noise								
	PCA	LDA	RPCA+LDA	DLRD	LatLRR	SRRS	DAE	Ours-I	Ours-II
C1	71.43 ± 1.12	47.77 ± 3.06	49.35 ± 1.55	82.96 ± 1.81	81.38 ± 1.25	86.45 ± 1.12	82.37 ± 1.37	89.77 ± 0.94	90.72 ± 1.25
C2	70.22 ± 1.56	45.89 ± 1.12	53.26 ± 1.84	60.46 ± 0.79	81.93 ± 0.92	82.03 ± 1.31	81.13 ± 0.83	89.52 ± 0.66	90.98 ± 0.54
C3	69.80 ± 0.65	36.42 ± 1.12	44.18 ± 2.65	49.88 ± 0.49	80.97 ± 0.45	82.05 ± 0.87	79.61 ± 1.02	86.97 ± 0.62	89.21 ± 0.92
C4	67.84 ± 0.83	27.13 ± 0.95	29.92 ± 0.96	41.52 ± 0.71	77.15 ± 0.72	79.83 ± 0.62	76.23 ± 0.59	84.57 ± 0.52	87.33 ± 0.87
C5	65.68 ± 0.76	16.79 ± 0.34	23.55 ± 0.46	73.82 ± 0.77	73.47 ± 0.62	74.95 ± 0.65	72.15 ± 0.60	83.64 ± 0.44	85.86 ± 0.62

Table 3. Recognition results (%) on CMU-PIE face database, where P1: {C02, C14}, P2: {C02, C27}, P3: {C14, C27}, P4: {C05, C07, C29}, P5: {C05, C14, C29, C34}, P6: {C02, C05, C14, C29, C31}. Red color denotes the best recognition rates. Blue color denotes the second best.

	Original images								
	PCA	LDA	RPCA+LDA	LatLRR	SRRS	LRCS	DAE	Ours-I	Ours-II
P1	69.03 ± 0.08	70.46 ± 0.05	74.39 ± 0.08	77.92 ± 0.03	78.27 ± 0.04	87.78 ± 0.02	85.65 ± 0.12	87.97 ± 0.06	88.04 ± 0.08
P2	69.21 ± 0.08	71.32 ± 0.02	75.55 ± 0.12	76.24 ± 0.12	78.74 ± 0.23	86.67 ± 0.01	84.32 ± 0.09	87.61 ± 0.03	87.88 ± 0.06
P3	68.52 ± 0.12	63.51 ± 0.75	75.29 ± 0.09	75.29 ± 0.07	77.45 ± 0.02	87.38 ± 0.19	84.53 ± 0.04	87.87 ± 0.09	88.01 ± 0.06
P4	52.65 ± 0.04	56.53 ± 0.02	61.17 ± 0.12	69.74 ± 0.05	71.44 ± 0.03	74.84 ± 0.04	71.87 ± 0.09	74.08 ± 0.07	75.06 ± 0.13
P5	34.94 ± 0.08	24.07 ± 0.25	38.66 ± 0.08	42.54 ± 0.12	38.86 ± 0.02	44.48 ± 0.03	42.32 ± 0.07	44.42 ± 0.10	45.35 ± 0.09
P6	29.09 ± 0.01	7.06 ± 0.01	31.94 ± 0.12	35.33 ± 0.04	30.16 ± 0.02	36.17 ± 0.01	33.50 ± 0.05	36.42 ± 0.03	36.54 ± 0.04
	Corrupted images with 10 % random noise								
	PCA	LDA	RPCA+LDA	LatLRR	SRRS	LRCS	DAE	Ours-I	Ours-II
P1	64.87 ± 0.32	26.71 ± 0.20	73.07 ± 0.11	73.10 ± 0.07	72.27 ± 0.05	78.98 ± 0.03	77.14 ± 0.11	81.02 ± 0.08	81.54 ± 0.07
P2	66.04 ± 0.08	23.19 ± 0.35	74.28 ± 0.12	73.24 ± 0.32	72.74 ± 0.18	78.67 ± 0.05	76.98 ± 0.06	81.12 ± 0.09	81.48 ± 0.10
P3	65.21 ± 0.04	20.34 ± 0.75	73.92 ± 0.12	73.85 ± 0.12	71.45 ± 0.08	78.38 ± 0.26	77.32 ± 0.09	81.94 ± 0.12	82.31 ± 0.08
P4	50.16 ± 0.04	46.72 ± 0.02	60.18 ± 0.14	58.94 ± 0.09	54.32 ± 0.03	65.84 ± 0.04	70.64 ± 0.08	73.73 ± 0.09	74.83 ± 0.12
P5	31.74 ± 0.08	6.67 ± 0.25	37.65 ± 0.09	39.26 ± 0.12	32.34 ± 0.02	39.48 ± 0.03	40.32 ± 0.09	43.92 ± 0.08	43.81 ± 0.09
P6	27.21 ± 0.01	4.06 ± 0.01	31.34 ± 0.06	32.07 ± 0.03	29.03 ± 0.02	32.57 ± 0.01	33.12 ± 0.09	35.33 ± 0.02	34.59 ± 0.07

In COIL dataset, we can observe that low-rank modeling based methods also achieve very good results compared with DAE, although the latter adds additive noises to train robust deep models. This demonstrates the robustness of low-rank modeling against noisy data. In the CMU-PIE dataset, DAE could achieve very similar performance to low-rank modeling based methods, in both supervised and unsupervised fashions. Similar results can be found from ALOI dataset. On CMU-PIE dataset, our algorithm cannot significantly improve the performance. One reason is that the facial appearances under different views on CMU-PIE dataset are very different. Considering additional illumination variations, this raises a very challenging feature learning problem on real-world dataset. However, our algorithm could still achieve promising performance, even better than a most recent multi-view learning method, LRCS. This further verifies the robustness of our algorithm against noises from real world. Generally, our supervised

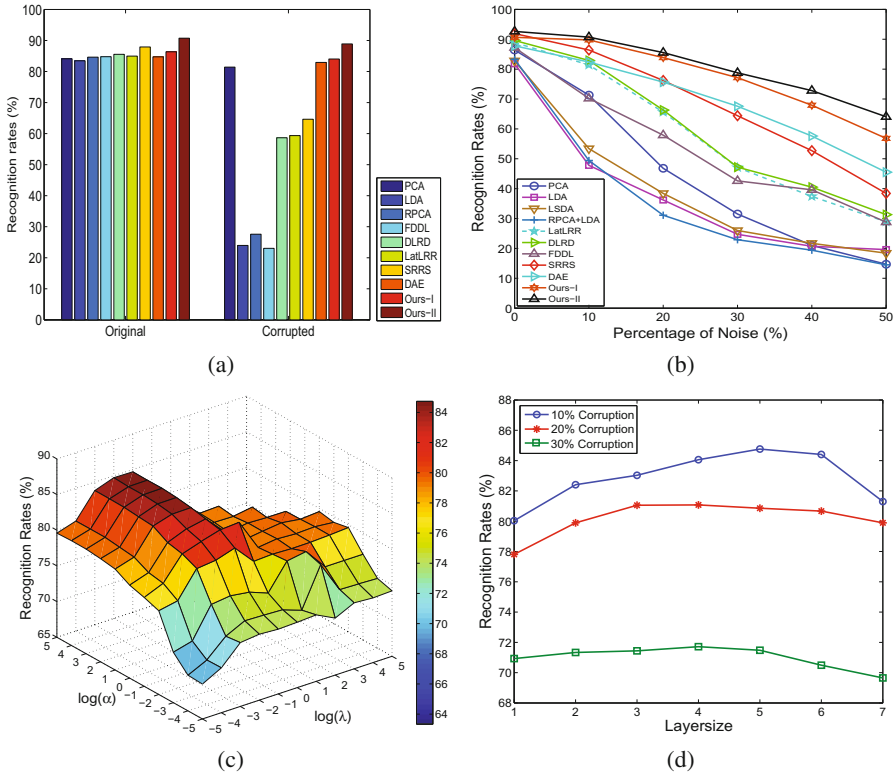

Fig. 4. (a) Recognition results (%) of 9 algorithms on ALOI-300 in original and 10 % corrupted cases. (b) Recognition rates of all comparisons on COIL database with different levels of noise. (c) Parameters analysis on α and λ. (d) The impact of layer size to the recognition performance.

model outperforms unsupervised one in almost all the cases. This demonstrates the importance of discriminative information in classification tasks.

4.4 Property Evaluation

In this section, we further evaluate several properties of our proposed algorithm, e.g., robustness to noise, parameter influence and layer size impact, to achieve a better understanding of the proposed model.

First of all, we evaluate the impacts of different corruption ratios to different algorithms. We evaluate 0 %, 10 %, 20 %, 30 %, 40 %, and 50 % corruptions with 20 objects on COIL dataset, and report results in Fig. 4(b), where our algorithm in two modes consistently outperforms other competitors. This demonstrates that our proposed algorithm can build a more robust feature extractor, especially for data with large corruption. Therefore, our algorithm could work efficiently in real-world applications with various noise.

Second, we conduct parameter analysis for our supervised model (Ours-II). Specifically, we evaluate the balance parameter λ and α for the low-rank dictionary and the graph terms, respectively. For better illustration, we jointly evaluate two parameters on corrupted COIL dataset with all 100 objects. Parameter influence results are listed in Fig. 4(c). From the results, we can notice larger value of α performs better especially when λ is small. Besides, we could see that small λ around 10^{-2} performs better. That is, the graph regularizer is more critical to our algorithm comparing to the low-rank constraint on dictionary. Without loss of generality, we set $\alpha = 10^2$ and $\lambda = 10^{-2}$ throughout our experiments.

Finally, we evaluate the impacts of layer size for Ours-II on corrupted COIL-100 with different corruptions (10 %, 20 %, 30 %). From Fig. 4(d), we can notice that our algorithm generally achieves better performance when the layer size goes up. That is, discriminative information is hopefully recovered by our deep encoding procedure. In other words, features would be refined from coarse to fine in a multi-layer fashion. However, we also observe that a much deeper structure would ruin the recognition performance. Therefore, in the experiments, we use a four-layer structure to generate the evaluation features.

5 Conclusion

In this paper, we developed a novel Deep Robust Encoder framework guided by a locality preserving low-rank dictionary learning scheme. Specifically, we designed a low-rank dictionary to constrain the output of the deep auto-encoder with corrupted input. In this way, the deep neural networks would generate more robust features by detecting noise from the corrupted data. Moreover, coefficient vectors z_i were maintained through the networks so that each output sample would be reconstructed by the most similar data samples in the dictionary with different weights. Furthermore, graph regularizers were developed to couple each layer's encoding to preserve more geometric structure. In experiments, we achieved more effective features for classification and results on several benchmarks demonstrated our method's superiority over other methods.

Acknowledgment. This research is supported in part by the NSF CNS award 1314484, ONR award N00014-12-1-1028, ONR Young Investigator Award N00014-14-1-0484, and U.S. Army Research Office Young Investigator Award W911NF-14-1-0218.

References

1. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: a deep convolutional activation feature for generic visual recognition. In: International Conference on Machine Learning, pp. 647–655 (2014)
2. Szegedy, C., Toshev, A., Erhan, D.: Deep neural networks for object detection. In: Neural Information Processing Systems, pp. 2553–2561 (2013)

3. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: closing the gap to human-level performance in face verification. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708. IEEE (2014)
4. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Neural Information Processing Systems, pp. 1097–1105 (2012)
5. Bengio, Y.: Learning deep architectures for ai. Found. Trends Mach. Learn. **2**(1), 1–127 (2009)
6. Le, Q.V., Ngiam, J., Coates, A., Lahiri, A., Prochnow, B., Ng, A.Y.: On optimization methods for deep learning. In: International Conference on Machine Learning, pp. 265–272 (2011)
7. Lee, C.Y., Xie, S., Gallagher, P., Zhang, Z., Tu, Z.: Deeply-supervised nets. In: International Conference on Artificial Intelligence and Statistics, pp. 562–570 (2015)
8. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
9. Hinton, G.E., Krizhevsky, A., Wang, S.D.: Transforming auto-encoders. In: Honkela, T., Duch, W., Girolami, M., Kaski, S. (eds.) ICANN 2011. LNCS, vol. 6791, pp. 44–51. Springer, Heidelberg (2011). doi:10.1007/978-3-642-21735-7_6
10. Droniou, A., Sigaud, O.: Gated autoencoders with tied input weights. In: International Conference on Machine Learning, pp. 154–162 (2013)
11. Kan, M., Shan, S., Chen, X.: Bi-shifting auto-encoder for unsupervised domain adaptation. In: IEEE International Conference on Computer Vision, pp. 3846–3854 (2015)
12. Ghifary, M., Bastiaan Kleijn, W., Zhang, M., Balduzzi, D.: Domain generalization for object recognition with multi-task autoencoders. In: IEEE International Conference on Computer Vision, pp. 2551–2559 (2015)
13. Wang, W., Arora, R., Livescu, K., Bilmes, J.: On deep multi-view representation learning. In: International Conference on Machine Learning, pp. 1083–1092 (2015)
14. Xia, C., Qi, F., Shi, G.: Bottom-up visual saliency estimation with deep autoencoder-based sparse reconstruction. IEEE Trans. Neural Netw. Learn. Syst. **27**(6), 1227–1240 (2016)
15. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J. Mach. Learn. Res. **11**, 3371–3408 (2010)
16. Wright, J., Ganesh, A., Rao, S., Peng, Y., Ma, Y.: Robust principal component analysis: exact recovery of corrupted low-rank matrices via convex optimization. In: Neural Information Processing Systems, pp. 2080–2088 (2009)
17. Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., Ma, Y.: Robust recovery of subspace structures by low-rank representation. IEEE Trans. Pattern Anal. Mach. Intell. **35**(1), 171–184 (2013)
18. Ding, Z., Fu, Y.: Low-rank common subspace for multi-view learning. In: IEEE International Conference on Data Mining, pp. 110–119. IEEE (2014)
19. Shao, M., Kit, D., Fu, Y.: Generalized transfer subspace learning through low-rank constraint. Int. J. Comput. Vis. **109**(1–2), 74–93 (2014)
20. Ding, Z., Shao, M., Fu, Y.: Deep low-rank coding for transfer learning. In: Twenty-Fourth International Joint Conference on Artificial Intelligence, pp. 3453–3459 (2015)
21. Jhuo, I.H., Liu, D., Lee, D., Chang, S.F., et al.: Robust visual domain adaptation with low-rank reconstruction. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2168–2175. IEEE (2012)

22. Ma, L., Wang, C., Xiao, B., Zhou, W.: Sparse representation for face recognition based on discriminative low-rank dictionary learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2586–2593. IEEE (2012)
23. Lin, Z., Chen, M., Ma, Y.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. arXiv preprint (2010). arXiv:1009.5055
24. Cai, J.F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**(4), 1956–1982 (2010)
25. Liu, D.C., Nocedal, J.: On the limited memory bfgs method for large scale optimization. Math. Program. **45**(1–3), 503–528 (1989)
26. Li, S., Fu, Y.: Learning robust and discriminative subspace with low-rank constraints. IEEE Trans. Neural Netw. Learn. Syst. PP(99), 1–13 (2015)
27. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cogn. Neurosci. **3**(1), 71–86 (1991)
28. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 711–720 (1997)
29. Liu, G., Yan, S.: Latent low-rank representation for subspace segmentation and feature extraction. In: IEEE International Conference on Computer Vision, pp. 1615–1622 (2011)