Jointly Non-negative, Sparse and Collaborative Representation for Image Recognition

Jun Xu[a], Zhou Xu[b], Wangpeng An[c,d], Haoqian Wang[c,d], David Zhang[e]

[a] *College of Computer Science, Nankai University, Tianjin, China*
[b] *School of Computer Science, Wuhan University, Wuhan, China*
[c] *Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China*
[d] *Shenzhen Institute of Future Media Technology, Shenzhen, China*
[e] *School of Science and Engineering, Chinese University of Hong Kong (Shenzhen), Shenzhen, China*

Abstract

Sparse representation (SR) and collaborative representation (CR) have been successfully applied in many image recognition tasks such as face recognition. In this paper, we propose a jointly Non-negative, Sparse and Collaborative Representation (NSCR) for image recognition. NSCR seeks a sparse and collaborative representation vector that represents each test image as a linear combination of training images, under a non-negative constraint. The non-negativity forces the SR and CR in our NSCR more discriminative for effective recognition. Based on the proposed NSCR, we propose a NSCR based classifier for image recognition. Extensive experiments on benchmark datasets demonstrate that the proposed NSCR classifier outperforms the previous SR or CR based approaches, as well as state-of-the-art deep ones, on diverse image recognition tasks. The code is available at `https://github.com/csjunxu/NSCR`.

Keywords: Non-negativity, sparse representation, collaborative representation, face recognition, digit recognition, object recognition, action recognition, fine-grained visual recognition

*Corresponding author is Jun Xu (nankaimathxujun@gmail.com).

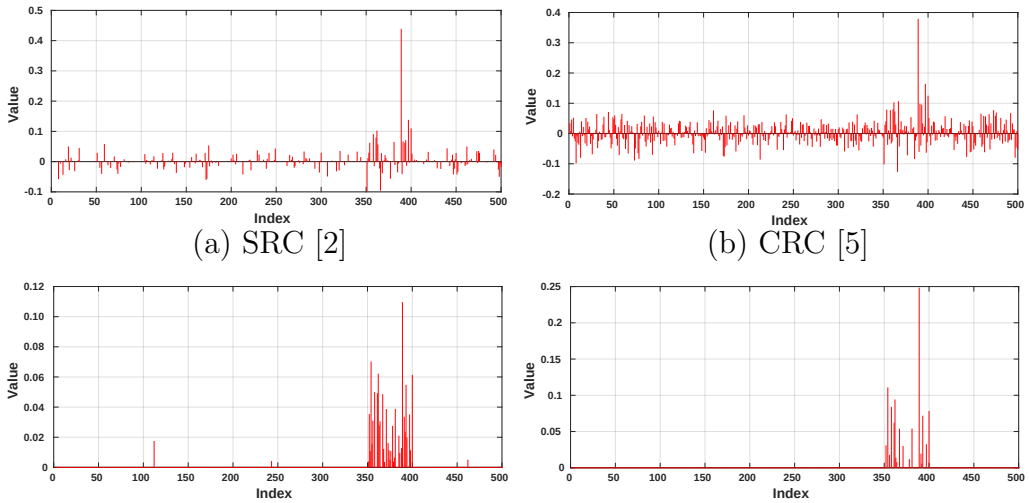
1. Introduction

Image recognition is a supervised learning problem in which the goal is to assign the test images to correct classes according to the labeled training images. It is a challenging pattern recognition problem with numerous applications such as object recognition [1], face recognition [2, 3], and action recognition [4], *etc*. Denote the training images as a data matrix $\boldsymbol{X} = [\boldsymbol{X}_1, ..., \boldsymbol{X}_K] \in \mathbb{R}^{D \times N}$ from K different classes. $N = \sum_{k=1}^{K} N_k$ is the number of whole training images. Each column of $\boldsymbol{X}$ is a vector reshaped from a training image. $\boldsymbol{X}_k \in \mathbb{R}^{D \times N_k}$ $(k = 1, ..., K)$ is the matrix consisting N_k training images from class k. The image recognition task aims to find the correct class that each test image $\boldsymbol{y} \in \mathbb{R}^D$ belongs to.

Numerous image recognition methods [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] have been proposed for face recognition, action recognition, and object recognition, *etc*. One important category is the representation based classifiers [2, 3, 4, 5, 6, 7, 8], which encode a test image $\boldsymbol{y}$ as a linear combination of the training images $\boldsymbol{X}$, and assign the test image $\boldsymbol{y}$ to the corresponding class with the minimum reconstruction error on $\boldsymbol{X}_k$ $(k = 1, ..., K)$. Three successive works in this category are Sparse Representation (SR) based Classifiers (SRC) [2], Collaborative Representation (CR) based Classifier (CRC) [5], and Non-negative Representation (NR) based Classifier (NRC) [4], which employ sparse, collaborative, or non-negative constraint on the representation of the test image over the training ones, respectively.

Despite the success of SRC/CRC/NRC classifiers on the image recognition task, they suffer from corresponding limitations. On one hand, both SRC and CRC would produce bias in the encoding coefficient vectors when the whole training images in matrix $\boldsymbol{X}$ are employed to reconstruct the test image $\boldsymbol{y}$. The reason is that, from the generative perspective, it is physically infeasible to reconstruct a real-world image from the training images with complex negaitive (subtraction) and positive (addition) coefficients [14]. Luckily, this problem can be naturally avoided by resorting to the non-negative (NR) constraint in NRC [4]. On the other, due to lack of proper regularization, NRC [4] is not flexible enough to deal with diverse real-world problems. Hence, NRC can be potentially promoted with the introduction of SR and (or) CR schemes.

With the above considerations, in this paper, we propose a jointly Non-negative, Sparse, and Collaborative Representation (NSCR) framework to integrate the benefits of non-negativity, sparsity, and collaborativity for image



(a) SRC [2] (b) CRC [5]

(c) NRC [4] (d) NSCR

Figure 1: An illustrative comparison of the coding vectors obtained by SRC [2], CRC [5], NRC [4], and the proposed NSCR schemes. The indices $1 \sim 500$ are ranged in a order of $[\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \boldsymbol{X}_9, \boldsymbol{X}_0]$, 50 indices for each $\boldsymbol{X}_i$. Since the digits "$3, 5, 0$" are similar to digit "8", NRC is prone to produce positive coefficients over $\boldsymbol{X}_3$ (indices $101 \sim 150$), $\boldsymbol{X}_5$ (indices $201 \sim 250$), and $\boldsymbol{X}_0$ (indices $451 \sim 500$). The proposed NSCR produces globally sparse, locally dense, and physically more reasonable representation than other schemes.

recognition. The proposed NSCR representation scheme is physically more meaningful and discriminative over previous sparse representation [2, 13], collaborative representation [5, 3], and non-negative representation [4]. In fact, the famous elastic net [15] is in fact the joint sparse and collaborative representation that linearly combines the ℓ_1 and ℓ_2 penalties of the lasso [16] and ridge regression models. Our work can be viewed as a further improvements of the elastic net [15] with an additional non-negative constraint. The major motivation of introducing non-negativity is that, in many real-world applications, the underlying signals represented by quantities can only take on non-negative values by nature. Examples validating this point include amounts of materials, chemical concentrations, and the compounds of end-members in hyperspectral images, just to name a few.

In Fig. 1, we illustrate the working mechanism of the proposed NSCR scheme through an example. We first select 50 images for each of the 10 handwritten digit numbers $0 \sim 9$ from the MNIST dataset [17]. The whole 500 images are ranged in a order of $[1, 2, ..., 9, 0]$, and formatted as a training data matrix $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \boldsymbol{X}_9, \boldsymbol{X}_0]$. We then randomly select a query image $\boldsymbol{y}$ of digit 8 from the test set of MNIST. For each image, we compute

a feature vector by using the scattering convolution network [18] (for more details, please refer to §4.5). Finally, we compute the coefficient vector of the query image $\boldsymbol{y}$ over the training data matrix $\boldsymbol{X}$ using SRC [2] (Fig. 1 (a)), CRC [5] (Fig. 1 (b)), NRC [4] (Fig. 1 (c)), and the propsoed NSCR (Fig. 1 (d)). We observe that the representation results proposed NSCR produces are globally sparse, locally dense, and physically more reasonable than those of other schemes.

Motivated by the advantages of the proposed NSCR representation scheme, we propose a simple yet effective NSCR based classifier for image recognition. The proposed NSCR classifier can be reformulated into a linear equality-constrained problem with two variables, and be solved under the alternating direction method of multipliers framework [19]. Each sub-problem can be solved efficiently in closed-form, and the convergence analysis is given to guarantee a suitable termination of the proposed algorithm. Extensive experiments on challenging benchmarks demonstrate that the proposed NSCR based classifier outperforms previous SR/CR/NR based classifiers on diverse image recognition tasks. The contribution of this paper are threefold:

- We propose a novel representation framework for image recognition problems. The proposed jointly non-negative, sparse and collaborative representation (NSCR) framework to integrate the benefits of non-negative, sparse, and collaborative representations.

- Based on the proposed NSCR representation framework, we develop a simple yet effective NSCR based classifier for image recognition.

- Extensive experiments on diverse benchmark datasets demonstrate that, the proposed NSCR based classifier outperforms previous sparse, collaborative, or non-negative representation based classifiers on face recognition, digit recognition, object recognition, action recognition, and fine-grained visual recognition, etc.

The remaining parts of this paper are organized as follows: In §2, we introduce the related work on representation based classifiers, which are closed related to ours. The proposed NSCR based classifier is formulated in §3. In §4, we compare NSCR with several state-of-the-art classifiers on diverse image recognition datasets. In §5, we conclude this paper.

2. Related Work on Representation based Classifiers

Nearest Neighbor based Classifier (NNC) [20] computes independently the distance between the test image and each of the training images, and assigns the label of the training image nearest to the test image as its predicted label. To reduce the computational costs of NNC, nearest subspace classifier (NSC) [8] computes the distance between the test image and each subspace, and then assigns the test image to its nearest subspace.

Sparse Representation based Classifier (SRC) [2] represents the test image as a linear combination of all the training images with ℓ_1 norm sparsity constraint imposed on the representational coefficients. SRC has shown promising recognition performance ever since its emergence. However, solving an ℓ_1 norm minimization problem usually requires huge computational costs and would be very slow in processing large-scale datasets.

Collaborative Representation based Classifier (CRC) [5] has been proposed to discuss the role of sparsity or collaborativity on face recognition. CRC represents the test image over all the training images collaboratively with an ℓ_2 norm constraint imposed on the representational coefficients, and demonstrates similar recognition accuracy while much faster speed than SRC. Inspired by the success of NSC and CRC, the collaborative representation optimized classifier (CROC) [6] is proposed to combine the advantages of NSC and CRC. A probabilistic collaborative representation based classifier (ProCRC) [3] is also proposed to compute the probability that a test image belongs to the collaborative subspace of all classes.

Non-negative Representation based Classifier (NRC) [4] is recently developed to investigate the usage of non-negative representation (NR) for image recognition. The NR scheme is inspired from the non-negative matrix factorization techniques [14, 21], and is very different from the ℓ_1 induced SR or the ℓ_2 induced CR schemes. NR can boost the representational power of homogeneous images while limiting the representational power of heterogeneous images, making the representation of the test image over training images sparse and discriminative simultaneously for effective recognition.

Extensions of SRC [2], CRC [5], and NRC [4] like ProCRC [3], CROC [6], and others [22, 23, 24, 25, 26, 27, 28] have been widely studied for various image recognition tasks such as face recognition, handwritten digit recognition, object recognition, and action recognition, etc. Among these methods, Yang *et al.* [23] analyzed the role of sparsity on pattern recognition. Cai *et al.* [3] extended CRC [5] into a probabilistic version for large scale recognition

tasks. Lin *et al.* [27] proposed a discriminative while comprehensive dictionary learning model for pattern recognition. All these methods extended the previous SRC [2], CRC [5], or NRC [4] significantly with huge improvements on model simplicity, computational speed, or recognition accuracy.

Formulation of SRC, CRC, and NRC. Now we formulate these representation based classifiers through optimization models. Assume that we have K classes of images, denoted by $\{\boldsymbol{X}_k\}, k = 1, ..., K$, where $\boldsymbol{X}_k \in \mathbb{R}^{D\times N_k}$ is the image matrix of class k. Each column of the matrix $\boldsymbol{X}_k$ is a training image from the k-th class. The whole training image matrix can be denoted as $\boldsymbol{X} = [\boldsymbol{X}_1, ..., \boldsymbol{X}_K] \in \mathbb{R}^{D\times N}$, where $N = \sum_{k=1}^{K} N_k$. Given a test image $\boldsymbol{y} \in \mathbb{R}^D$, both SRC and CRC compute the coding vector $\boldsymbol{c}$ of $\boldsymbol{y}$ over the training images $\boldsymbol{X}$, by solving the following minimization problem:

$$\min_{\boldsymbol{c}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 + \lambda\|\boldsymbol{c}\|_p^q, \tag{1}$$

where $p = 0, 1$, $q = 1$ for SRC and $p = q = 2$ for CRC, and λ is the regularization parameter. Different from SRC/CRC, NRC computes the coding vector $\boldsymbol{c}$ of $\boldsymbol{y}$ over $\boldsymbol{X}$ by solving the following minimization problem:

$$\min_{\boldsymbol{c}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 \quad \text{s.t.} \quad \boldsymbol{c} > 0. \tag{2}$$

The obtained coding vector $\hat{\boldsymbol{c}}$ can be written as $\hat{\boldsymbol{c}} = [\hat{\boldsymbol{c}}_1^\top, .., \hat{\boldsymbol{c}}_K^\top]^\top$, where $\hat{\boldsymbol{c}}_k, k = 1, ..., K$ is the coding sub-vector of $\boldsymbol{y}$ over the sub-matrix $\boldsymbol{X}_k$ of images in class k. Assume that the test image $\boldsymbol{y}$ belongs to the k-th class, then it is highly possible that $\boldsymbol{X}_k\hat{\boldsymbol{c}}_k$ can be a good approximation of the test image $\boldsymbol{y}$, i.e., $\boldsymbol{y} \approx \boldsymbol{X}_k\hat{\boldsymbol{c}}_k$. Therefore, SRC, CRC, and NRC first compute the approximation residual of $\boldsymbol{y}$ in each class as:

$$\text{label}(\boldsymbol{y}) = \arg\min_k \{\|\boldsymbol{y} - \boldsymbol{X}_k\boldsymbol{c}_k\|_2\}. \tag{3}$$

Then the class (e.g., k) with minimal residual would be the predicted class for $\boldsymbol{y}$. The major difference among SRC, CRC, and NRC lies in the coding vector $\boldsymbol{c}$. For SRC, the coding vector $\boldsymbol{c}$ is sparse induced by the ℓ_1 norm, but not likely to be non-zero in the correct class. For CRC, $\boldsymbol{c}$ is generally dense over all classes, this is the collaborative property induced by the ℓ_2 norm. For NRC, $\boldsymbol{c}$ is non-negative and the significant coefficients are mostly fall into class $\boldsymbol{c}_k$, in which the images are most similar to the test image $\boldsymbol{y}$. The overall recognition framework of the SRC/CRC/NRC classifiers is summarized in Algorithm 1.

Algorithm 1: The SRC/CRC or NRC Algorithms

Input: Training sample matrix $\boldsymbol{X} = [\boldsymbol{X}_1, ..., \boldsymbol{X}_K]$, query sample $\boldsymbol{y}$;

1. Normalize the columns of $\boldsymbol{X}$ to have unit ℓ_2 norm;
2. Compute the coding vector $\hat{\boldsymbol{c}}$ of $\boldsymbol{y}$ over $\boldsymbol{X}$ via Eqn. (1) (SRC/CRC) or Eqn. (2) (NRC);
3. Compute the approximation residuals $r_k = \|\boldsymbol{y} - \boldsymbol{X}_k\hat{\boldsymbol{c}}_k\|_2$;

Output: Label of $\boldsymbol{y}$: $\text{label}(\boldsymbol{y}) = \arg\min_k\{r_k\}$.

In this work, we introduce a jointly Non-negative, Sparse and Collaborative Representation (NSCR) to integrate the benefits of the widely studied SR/CR/NR schemes for image recognition. Extensive experiments demonstrate the effectiveness of the proposed NSCR based classifier on diverse image recognition datasets.

3. Non-negative Sparse and Collaborative Representation based Classifier

In this section, we introduce the proposed jointly Non-negative, Sparse and Collaborative Representation (NSCR) model, and develop a NSCR model based classifier for image recognition.

3.1. Non-negative, Sparse and Collaborative Representation (NSCR)

Given a test image $\boldsymbol{y}$ and the training image matrix $\boldsymbol{X}$, the proposed NSCR model is formulated as follows:

$$\min_{\boldsymbol{c}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 + \alpha\|\boldsymbol{c}\|_2^2 + \beta\|\boldsymbol{c}\|_1 \quad \text{s.t.} \quad \boldsymbol{c} \geq 0, \tag{4}$$

where $\boldsymbol{c}$ is the coding vector of the test image $\boldsymbol{y}$ over the training image matrix $\boldsymbol{X}$, α, β are regularization parameters. Here, we integrate the regularization terms of SRC [2] and CRC [5], as well as the additional constraint of NRC [4], into our NSCR model. The integrated regularization linearly combines the ℓ_1 and ℓ_2 penalties, and thus enjoys robust and flexibile property as the elastic net [15]. As we will see in the next section, the additional non-negative constraint in the proposed NSCR model make the obtained representation more sparse and discriminative for effective recognition. Since $\boldsymbol{c} > 0$ is non-negative , we have $\|\boldsymbol{c}\|_1 = \boldsymbol{1}^\top\boldsymbol{c}$. Then Eqn. (4) is equivalent to

$$\min_{\boldsymbol{c}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 + \alpha\|\boldsymbol{c}\|_2^2 + \beta\boldsymbol{1}^\top\boldsymbol{c} \quad \text{s.t.} \quad \boldsymbol{c} \geq 0, \tag{5}$$

The proposed NSCR model (5) only contains two explicit parameters (α, β), and hence especially easy to be solved (as will be demonstrated in §3.2). The proposed NSCR can be viewed implicitly as a non-negative constrained elastic LASSO model [15], which provides an intuitive explanation for the experimental findings that it achieves better performance than all previous classifiers such as SRC [2], CRCs [3, 5, 6], and NRC [4] (please refer to §4 for more details).

Discussion. There are several advantages in introducing non-negativity into the popular sparse and collaborative representation schemes: 1) Non-negativity can brings better discriminative ability over SR/CR with higher recognition accuracy. This point is validated by the non-negative representation based classifier [4] that, a non-negative representation of the test image could be positive over the images from the homogeneous class while zero over the images from heterogeneous classes. In SR/CR based classifiers [2, 5, 3], the test image can be approximated by linear combinations of the whole training images. Though mathematically feasible, it is physically problematic to reconstruct an image with complex subtractions and additions in real-world applications [14]. 2) It is more reasonable to utilize non-negativity with the biological modeling of the visual data and often lead to better performance for data representation [14]. The non-negativity property allows only non-negative combination of multiple training images to additively reconstruct the test image, which is also compatible with the intuitive notion of combining parts into a whole [14]. 3) Non-negativity helps automatically extract sparse and interpretable image representation to reconstruct the test image [29, 30]. For example, the authors in [29] compared the non-negative least square (NNLS) model with the non-negative LASSO [16] on sparse recovery, and found that the NNLS model could achieve similar or even better performance than non-negative LASSO.

3.2. Optimization

Since the proposed NSCR model (5) does not have an analytical solution, we employ the variable splitting method [31] to solve it. Specifically, we firstly reformulate the NSCR model (5) into a linear equality-constrained problem by introducing an auxiliary variable $\boldsymbol{z}$:

$$\min_{\boldsymbol{c},\boldsymbol{z}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 + \alpha\|\boldsymbol{c}\|_2^2 + \beta\mathbf{1}^\top\boldsymbol{c} \quad \text{s.t.} \quad \boldsymbol{z} = \boldsymbol{c}, \boldsymbol{z} \geq 0. \tag{6}$$

Then, the alternating direction method of multipliers(ADMM) [19] algorithm is employed to solve the NSCR model (5).

Algorithm 1: Solve NSCR (6) via ADMM

Input: Test image $\boldsymbol{y}$, training image matrix $\boldsymbol{X}$,
tolerance value Tol > 0, maximal iteration number T,
regularization parameters α, β, and penalty parameter $\rho > 0$;

Initialization: $\boldsymbol{z}_0 = \boldsymbol{c}_0 = \boldsymbol{\delta}_0 = \mathbf{0}$, $t = 0$;

for $t = 0 : T - 1$ **do**

1. Update $\mathbf{c}_{t+1}$ as $\boldsymbol{c}_{t+1} = (\boldsymbol{X}^\top \boldsymbol{X} + \frac{2\alpha+\rho}{2}\boldsymbol{I})^{-1}(\boldsymbol{X}^\top \boldsymbol{y} + \frac{\rho}{2}\boldsymbol{z}_t + \frac{1}{2}\boldsymbol{\delta}_t - \frac{\beta}{2})$;
2. Update $\mathbf{z}_{t+1}$ as $\boldsymbol{z}_{t+1} = \max(0, \boldsymbol{c}_{k+1} - \rho^{-1}\boldsymbol{\delta}_t)$;
3. Update $\boldsymbol{\delta}_{t+1}$ as $\boldsymbol{\delta}_{t+1} = \boldsymbol{\delta}_t + \rho(\boldsymbol{z}_{t+1} - \boldsymbol{c}_{t+1})$;

if (Convergence condition is satisfied)

4. Stop;

end if

end for

Output: Coding vectors $\boldsymbol{z}_T$ and $\boldsymbol{c}_T$.

$$\mathcal{L}(\boldsymbol{c}, \boldsymbol{z}, \boldsymbol{\delta}, \lambda, \rho) = \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 + \alpha\|\boldsymbol{c}\|_2^2 + \beta\mathbf{1}^\top\boldsymbol{c} + \langle\boldsymbol{\delta}, \boldsymbol{z} - \boldsymbol{c}\rangle + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{c}\|_2^2, \quad (7)$$

where $\boldsymbol{\delta}$ is the augmented Lagrangian multiplier and $\rho > 0$ is the penalty parameter. We initialize the vector variables $\boldsymbol{c}_0$, $\boldsymbol{z}_0$, and $\boldsymbol{\delta}_0$ to be zero vector and set $\rho > 0$ with a suitable value. By taking derivatives of the Lagrangian function and setting the derivative function to be zero, we can iteratively update the variables. The specific updating process is as follows:

(1) **Updating $\boldsymbol{c}$ while fixing $\boldsymbol{z}$ and $\boldsymbol{\delta}$**:

$$\min_{\boldsymbol{c}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 + \alpha\|\boldsymbol{c}\|_2^2 + \beta\mathbf{1}^\top\boldsymbol{c} + \frac{\rho}{2}\|\boldsymbol{c} - (\boldsymbol{z}_t + \rho^{-1}\boldsymbol{\delta}_t)\|_2^2. \quad (8)$$

This is an elastic net [15] problem without explicit ℓ_1 regularization, and hence can be solved with a closed-form solution:

$$\boldsymbol{c}_{t+1} = (\boldsymbol{X}^\top \boldsymbol{X} + \frac{2\alpha+\rho}{2}\boldsymbol{I})^{-1}(\boldsymbol{X}^\top \boldsymbol{y} + \frac{\rho}{2}\boldsymbol{z}_t + \frac{1}{2}\boldsymbol{\delta}_t - \frac{\beta}{2}) \quad (9)$$

The Eqn. (9) is slow when the number of images N in $\boldsymbol{X}$ is much larger than their feature dimension D. In order to make the proposed NSCR classifier scalable to large scale visual datasets, we employ the well-known Woodbury Identity Theorem to reduce the computational cost of the inversion operation in Eq. (9). Specifically, the update of $\boldsymbol{c}$ in (9) can be computed as follows:

$$\begin{aligned}\boldsymbol{c}_{t+1} =&(\frac{2}{2\alpha+\rho}\boldsymbol{I}-(\frac{2}{2\alpha+\rho})^2\boldsymbol{X}^\top(\boldsymbol{I}+\frac{2}{2\alpha+\rho}\boldsymbol{X}\boldsymbol{X}^\top)^{-1}\boldsymbol{X})\\&\times(\boldsymbol{X}^\top\boldsymbol{y}+\frac{\rho}{2}\boldsymbol{z}_t+\frac{1}{2}\boldsymbol{\delta}_t-\frac{\beta}{2}).\end{aligned} \tag{10}$$

By this step, the complexity of updating $\boldsymbol{c}$ is reduced from $\mathcal{O}(N^3)$ to $\mathcal{O}(DN^2)$. Since $(\boldsymbol{X}^\top\boldsymbol{X}+\frac{\rho}{2}\boldsymbol{I})^{-1}$ is not updated during iterations, we can also pre-compute it and store it before iterations. This strategy further saves plentiful computational costs.

(2) **Updating $\boldsymbol{z}$ while fixing $\boldsymbol{c}$ and $\boldsymbol{\delta}$**:

$$\min_{\boldsymbol{z}}\|\boldsymbol{z}-(\boldsymbol{c}_{t+1}-\rho^{-1}\boldsymbol{\delta}_t)\|_2^2 \quad \text{s.t.} \quad \boldsymbol{z}\geq 0. \tag{11}$$

The solution of $\boldsymbol{Z}$ is

$$\boldsymbol{z}_{t+1}=\max(0,\boldsymbol{c}_{t+1}-\rho^{-1}\boldsymbol{\delta}_t)), \tag{12}$$

where the "max" function operates element-wisely.

(3) **Updating the Lagrangian multiplier $\boldsymbol{\delta}$**:

$$\boldsymbol{\delta}_{t+1}=\boldsymbol{\delta}_t+\rho(\boldsymbol{z}_{t+1}-\boldsymbol{c}_{t+1}). \tag{13}$$

We repeat these alternative updating steps, until some certain convergence condition is reached (or the number of iterations exceeds a preset threshold T). In this work, the convergence condition of ADMM algorithm is reached when $\|\boldsymbol{z}_{t+1}-\boldsymbol{c}_{t+1}\|_2\leq \text{Tol}$, $\|\boldsymbol{c}_{t+1}-\boldsymbol{c}_t\|_2\leq \text{Tol}$, and $\|\boldsymbol{z}_{t+1}-\boldsymbol{z}_t\|_2\leq \text{Tol}$ are simultaneously satisfied. The $\text{Tol}>0$ is a very small positive value. Note that the objective function and constraints of NSCR (6) are all strictly convex, therefore the problem (6) solved by the ADMM algorithm could be guaranteed to converge to a global optimum. The procedures of solving our NSCR model are summarized in Algorithm 1.

Convergence. The convergence of Algorithm 1 can be guaranteed since the overall objective function (13) is convex with a global optimum. In Figure 2, we plot the convergence curves of the errors of $\|\mathbf{c}_{t+1}-\mathbf{z}_{t+1}\|_2$, $\|\mathbf{z}_{t+1}-\mathbf{z}_t\|_2$, $\|\mathbf{c}_{t+1}-\mathbf{c}_t\|_2$. One can see that they all approach to 0 in 100 iterations. For speed consideration, we set the iteration number as $T=20$.

3.3. The NSCR based Classifier

Denote by $\boldsymbol{y}\in\mathbb{R}^D$ a test image and $\boldsymbol{X}\in\mathbb{R}^{D\times N}=[\boldsymbol{X}_1,...,\boldsymbol{X}_K]$ the training image matrix, where $\boldsymbol{X}_k\in\mathbb{R}^{D\times N_k}$ contains the training images from

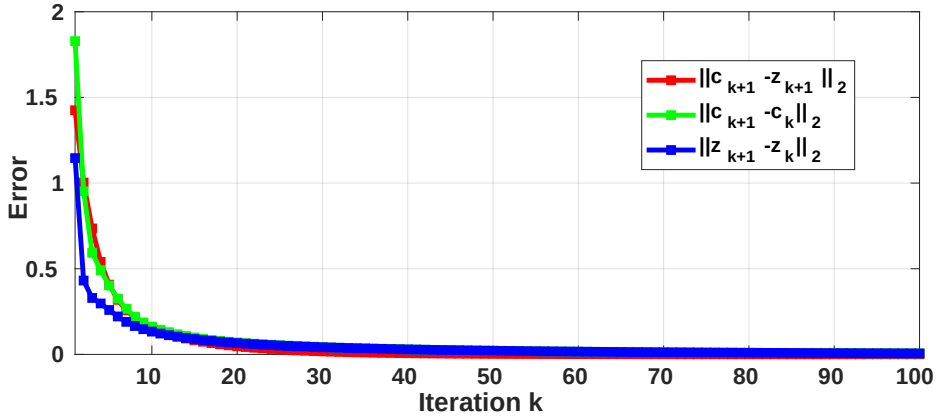


Figure 2: The convergence curves of $\|\mathbf{c}_{t+1} - \mathbf{z}_{t+1}\|_2$ (red line), $\|\mathbf{z}_{t+1} - \mathbf{z}_t\|_2$ (blue line), and $\|\mathbf{c}_{t+1} - \mathbf{c}_t\|_2$ (green line) of the proposed NSCR model (6) on Extended Yale B [32].

Algorithm 2: The NSCR based Classifier

Input: Test image $\boldsymbol{y}$, training image matrix $\boldsymbol{X} = [\boldsymbol{X}_1, ..., \boldsymbol{X}_K]$, tolerance value Tol > 0, maximal iteration number T, regularization parameters α, β, and penalty parameter $\rho > 0$;

1. Normalize the columns of $\boldsymbol{X}$ to have unit ℓ_2 norm;
2. Encode $\boldsymbol{y}$ over $\boldsymbol{X}$ by solving the NSCR model (6) via Algorithm 1: $\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{c}\|_2^2 + \alpha\|\boldsymbol{c}\|_2^2 + \beta\boldsymbol{c} \quad \text{s.t.} \quad \boldsymbol{c} \geq 0$;
3. Compute the regularized residuals via $r_k = \|\boldsymbol{y} - \boldsymbol{X}_k\hat{\boldsymbol{c}}_k\|_2$;

Output: Identity$(\boldsymbol{y}) = \arg\min_k\{r_k\}$.

class k. We first apply unit ℓ_2 normalization on $\boldsymbol{y}$ and each column of $\boldsymbol{X}$, and then compute the coding vector $\hat{\boldsymbol{c}}$ according to the NSCR model (6). Next we compute the class representation residual $\|\boldsymbol{y} - \boldsymbol{X}_k\hat{\boldsymbol{c}}_k\|_2$ and determine its recognition, where $\hat{\boldsymbol{c}}_k$ is the coding sub-vector corresponding to the class k. The proposed NSCR based Classifier is summarized in Algorithm 2.

3.4. Complexity Study

In the iterative process to solve the NSCR model via ADMM, the cost of updating $\boldsymbol{c}$ is $\mathcal{O}(DN^2)$ by employing the Woodbury Identity Theorem [33]. The cost of updating $\boldsymbol{z}$ is $\mathcal{O}(D)$. The costs of updating $\boldsymbol{\delta}$ is also $\mathcal{O}(D)$. So the overall complexity of Algorithm 1 is $\mathcal{O}(DN^2T)$. In Algorithm 2, since the costs of computing the residuals can be ignored, the overall cost of Algorithm 2 is $\mathcal{O}(DN^2T)$.

4. Experiments

In this section, we evaluate the proposed jointly Non-negative, Sparse and Collaborative Representation (NSCR) based Classifier, and compare it with state-of-the-art classifiers on diverse image recognition datasets.

4.1. Parameter Settings

The proposed NSCR approach is solved by ADMM and has five parameters: the regularization parameters α, β, the iteration number T, the penalty parameter ρ, and the small tolerance value Tol. In all experiments, we set $T = 20$, $\rho = 10$, Tol $= 0.001$, and determine the regularization parameters α, β by 5-fold cross validation on the training subset of each dataset. For the comparison methods, we use their source codes provided by the corresponding authors, and tune their parameters to achieve their best recognition accuracy on different datasets.

4.2. Datasets

Face Recognition. We first introduce two commonly tested datasets for face recognition, i.e., **AR** face dataset [34] and **Extend Yale B** dataset [32]. The **AR** face dataset [34] is consisted of more than 4,000 face images of 126 subjects, including 56 women and 70 men. The contents of these images are frontal view faces with diverse facial expressions, illumination environments, and partial occlusions (such as sun glasses and scarf). The images were taken under strictly controlled environments. There is no restriction on wear (clothes, glasses, etc.), make-up, or hair style imposed to participants. The same images of each participated person were taken in two sessions, separated by two weeks. The **Extend Yale B** dataset [32] has totally 2,432 face images, which are taken from 38 participated persons. The 64 near frontal images (in gray-scale) per person are taken under different illumination environments. The original images are of size 192×168, and we resize them to size 54×48 in our experiments for dimension reduction purposes.

Handwritten Digit Recognition. We then introduce two commonly tested datasets for handwritten digit recognition, i.e., **USPS** dataset [35] and **MNIST** dataset [17]. The **USPS** dataset [35] contains 9,298 images for handwritten digit numbers of $\{0, 1, ..., 9\}$. The training set contains 7,291 images while the testing set contains the other 2,007 images. Each image is of 16×16 grayscale pixels. **MNIST** dataset [17] of handwritten digits contains 60,000 training images and 10,000 testing images for digit numbers of $\{0, 1, ..., 9\}$.

The digit in each image has been size-normalized and centered in a fix-size image of 28×28 grayscale pixels.

Human Action Recognition. We also introduce one widely tested dataset for human action recognition, i.e., the **Stanford 40 Actions** dataset [36]. **Stanford 40 Actions** dataset [36] contains 40 different classes of human actions, e.g., brushing teeth, cleaning the floor, reading book, and throwing a Frisbee, etc. It contains totally 9,352 images, $180 \sim 300$ images per action.

Object Recognition. The widely tested dataset for object recognition we used in this work is the **Caltech-256** dataset [37]. It is consisted of 256 categories of objects, in which there are at least 80 images for each category. This large dataset has a total number of 30,608 images.

Fine-grained Visual Recognition. Finally, we introduce four widely tested benchmarks for fine-grained visual recognition, i.e., **Caltech-UCSD Birds (CUB200-2011)** dataset [38], **Oxford 102 Flowers** dataset [39], **Aircraft** dataset [40], and **Cars** dataset [41]. The **Caltech-UCSD Birds (CUB200-2011)** dataset [38] includes 11,788 images of 200 different birds. Each bird class has around 60 images. The challenging of this dataset for visual recognition comes from the distinct variations in illumination, pose, and viewpoint in each bird class. The **Oxford 102 Flowers** dataset [39] includes 8,189 images of 102 different flowers. Each flower class has more than 40 images. These flower images are captured under diverse lighting conditions, flower poses, and image scales. The difficulty of this dataset comes from the fact that there exist large variations within the same flower but small differences across different flowers. The **Aircraft** dataset [40] includes 10,000 images of 100 different aircraft model variants, 100 images for each. These aircrafts appear at diverse appearances, scales, and design structures, making this dataset very challenging for visual recognition. The **Cars** dataset [41] includes 16,185 images of 196 car classes. Each car class contains around 80 images at different scales and heavy clutter background, making this dataset very challenging for visual recognition.

4.3. Comparison Methods

We compare the proposed NSCR with several representation based classifiers such as SRC [2], CRC [5], CROC [6], ProCRC [3], and NRC [4]. We also compare NSCR with linear Support Vector Machine (SVM) [42], a widely used discriminative classifier. In §4.4, we compare these competing methods on face recognition via the **AR** [34] and **Extended Yale B** [32]. In §4.5, we compare these methods on digit recognition through the **USPS** [35] and

MNIST [17] datasets. We also compare these methods on action recognition by using **Stanford 40 Actions** dataset [36] in §4.6, on object recognition by using **Caltech-256** dataset [37] in §4.7, and on four challenging fine-grained visual recognition tasks by using the **Caltech-UCSD Birds** (**CUB-200-2011**) [38], **Oxford 102 Flowers** [39], **Aircraft** [40], and **Cars** [41] datasets in §4.8. On fine-grained visual recognition tasks, we also compare with some state-of-art deep method such as Symbiotic [43], FV-FGC [28], and B-CNN [44].

4.4. Results on Face Recognition

For the **AR** dataset [34], as suggested in [2, 5, 4], we select a subset containing 50 female subjects and 50 male subjects, with only changes on expression and illumination, in experiments. For each subject, the 7 images from Session 1 are used as the training set, while the other 7 images from Session 2 are used as the test set. All these images are cropped to size of 60×43, followed by a normalization operation with unit ℓ_2 norm. For further acceleration, we project the vectors of these images to dimension $d = 54, 120, 300$. The results on recognition accuracy (%) of the comparison methods are summarized in Table 1. We observe that the proposed NSCR classifier achieves the highest accuracy results on all cases of projected dimensions, i.e., when d is 54, 120, or 300. The parameters α, β in our NSCR on this dataset are set as $\alpha = \beta = 0.01$.

d	**SVM** [42]	**SRC** [2]	**CRC** [5]	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
54	81.6	82.1	80.3	82.0	81.4	86.0	**87.3**
120	89.3	88.3	90.0	90.8	90.7	91.3	**92.1**
300	91.6	90.3	93.7	93.7	93.7	94.0	**94.7**

Table 1: Recognition accuracy (%) of the comparison classifiers on the **AR** dataset [34]. The images are projected onto dimension d by PCA.

For the **Extended Yale B** dataset, as suggested in [2, 5, 4], the original images are of size 192×168. We resize the images to size 54×48 and normalize the images to be in an ℓ_2 unit ball. The images are projected onto a d-dimensional subspace by using PCA. Just as suggested in [5], we randomly split this dataset into two halves. Each half contains 32 images of every person. We use one half dataset as the training set, and the other half as the test set. For further acceleration, we project these images to smaller dimensions as $d = 84, 150$, or 300. The results on recognition accuracy (%) of the

comparison methods are summarized in Table 2. We observe that the proposed NSCR classifier achieves better performance than all the comparison methods on all cases of $d = 84, 120, 300$. The parameters α, β in our NSCR on this dataset are set as $\alpha = 0.05$ and $\beta = 0.01$.

d	**SVM** [42]	**SRC** [2]	**CRC** [5]	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
84	93.4	95.5	95.0	95.5	93.4	95.6	**97.0**
150	95.8	96.9	96.3	97.1	95.3	97.1	**98.6**
300	96.9	97.7	97.9	98.2	96.2	98.2	**99.7**

Table 2: Recognition accuracy (%) of the comparison classifiers on the **Extended Yale B** dataset [32]. The results are averaged on 10 independent trials. The images are projected onto dimension d by PCA.

4.5. Results on Handwritten Digit Recognition

For the **USPS** dataset [35], as suggested in [3], we randomly select $N =$ 50, 100, 200, and 300 images from each digit class of the training set as the training images, and use all the samples in the test set as the test images. We run the experiments 10 times and provide the mean results. In Table 3, we summarize the recognition accuracy (%) of the comparison methods. We observe that the proposed NSCR classifier outperforms all the comparison methods on all cases of selecting $N = 50, 100, 200, 300$ images from each class as the training set. With the increasing of the number of training images, the recognition accuracy of all the comparison methods increases consistently, including the proposed NSCR. However, we observe that the ProCRC will not perform better when the number of training images increases from 200 to 300, while the proposed NSCR approach can still increase a lot from 95.3% to 95.7%. Similar trends can also be found for the other classifiers. The parameters α, β in our NSCR on this dataset are set as $\alpha = 0.01$ and $\beta = 0.05$.

N	**SVM** [42]	**SRC** [2]	**CRC** [5]	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
500	91.6	91.4	89.2	91.9	90.9	92.3	**93.1**
1,000	92.5	93.1	90.6	91.3	91.9	93.7	**94.4**
2,000	93.1	94.2	91.4	91.7	92.2	94.6	**95.3**
3,000	93.2	94.8	91.5	91.8	92.2	95.1	**95.7**

Table 3: Recognition accuracy (%) of the comparison classifiers as a function to the number (N) of selected images from each class for training, on the **USPS** dataset [35]. The results are averaged on 10 independent runs.

For the **MNIST** dataset [17], we resize each image into the size 16×16 as suggested in [3]. We randomly selected $N = 50, 100, 300, 500$ images from each class of the training set to construct the training images, and use all the samples in the test set as test images. Different from [3], for each image in the MNIST dataset, in our NSCR we compute a feature vector by using the scattering convolution network (SCN) [18]. The feature vector is a concatenation of coefficients in each layer of the SCN network, and is deformation stable and translation invariant. Each feature vector is of size 3,472. The feature vectors for all images are then projected onto a 500-dimensional space by using PCA. The subspace clustering techniques [45, 46] are then applied to the projected features. We run the experiments 10 times and provide the mean recognition accuracy (%). In Table 4, we summarize the recognition accuracy (%) of all the comparison classifiers. We observe that the proposed NSCR classifier achieves comparable or better performance when compared to the other classifiers, no matter how many images (50, 100, 200, or 300) from each class are selected as the training images. With the increasing of the training images, the recognition accuracy of all the competing classifers increases consistently. Due to the advanced features are used by employing the SCN network [18], the recognition accuracy of the comparison methods are consistently higher than the corresponding results provided in [3]. The parameters α, β in our NSCR on this dataset are set as $\alpha = \beta = 0.05$.

N	**SVM** [42]	**SRC** [2]	**CRC** [5]	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
500	97.4	95.6	97.4	97.0	97.2	97.8	**98.6**
1,000	98.1	96.8	98.3	98.3	98.0	98.3	**98.9**
3,000	98.5	97.9	98.7	98.7	98.5	98.8	**99.3**
6,000	98.6	98.0	98.8	98.8	98.6	99.0	**99.4**

Table 4: Recognition accuracy (%) of the comparison classifiers with respect to the number (N) of selected samples from each class for training, on the **MNIST** dataset [17]. The results are averaged on 10 independent runs. The images are projected onto a 500-dimensional space by using PCA.

4.6. Results on Action Recognition

For the **Stanford 40 Actions** dataset [36], as suggested by [3], we follow the training-test split settings scheme of [36], and randomly select 100 images from each class as the training images and employ the remaining images as the testing set. Similar as [3], we employ two different types of features to

demonstrate the effectiveness of the proposed NSCR classifier. The first type of features are the Bag-of-Words features based on SIFT [47] obtained by employing the VLFeat library [48], hence referred as BOW-SIFT feature. The sizes of image patch and stride are set as 16×16 and 8 pixels, respectively. The codebook is trained by the k-means algorithm, with a feature size of 1,024. We use a 2-level spatial pyramid representation [9]. The final feature dimension of each image is 5,120. The second type of features are the CNN features extracted by using the VGG-verydeep-19 [49], and therefore referred as VGG19 features. We employ the activations of the penultimate layer as local features, extracted from 5 scales of $\{2^s, s = -1, -0.5, 0, 0.5, 1\}$. We also pool all local features together, regardless of locations and scales. The final dimension of the feature for each image is 4,096. Note that both the BOW-SIFT features and VGG19 features are ℓ_2 normalized.

The results of recognition accuracy (%) by the comparison methods are summarized in Table 5. We observe that the proposed NSCR classifier achieves higher accuracy than previous representation based classifiers, e.g., SRC [2], CRC [5], and NRC [4], on both types of features. This demonstrates that the proposed jointly NSCR representation is indeed more effective than previous SR, CR, and NR representation schemes. The parameters α, β in our NSCR on this dataset are set as $\alpha = 0.05$ and $\beta = 0.1$.

Algorithms	**Softmax**	**M-SVM** [42]	**SRC** [2]	**CRC** [5]
VGG19	77.2	79.0	78.7	78.2
BOW-SIFT	21.1	24.0	24.2	24.6
Algorithms	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
VGG19	79.2	80.9	81.9	**82.3**
BOW-SIFT	24.5	28.4	29.2	**29.5**

Table 5: Recognition accuracy (%) of the comparison classifiers on the **Stanford 40 Actions** dataset [36].

4.7. Results on Object Recognition

For the **Caltech-256** dataset [37], as suggested in [3, 4], we randomly select 15, 30, 45, and 60 images from each object class to form the training set, respectively, and use the remaining images as the test set. We run the proposed NSCR and the comparison methods 10 times, each time for a random partition. We only report the mean recognition accuracy (%). Similar to the operations on the **Stanford 40 Actions** dataset [36] mentioned above,

we employ two different types of features, i.e., the BoW-SIFT features extracted from [47] and the VGG19 features extracted from VGG-verydeep-19 [49]. The results on recognition accuracy (%) are summarized in Table 6. We observe that the proposed NSCR classifier achieves better performance than the other competing methods. The parameters α, β in our NSCR on this dataset are set as $\alpha = 0.01$ and $\beta = 0.05$.

Algorithms	**Softmax**	**M-SVM** [42]	**SRC** [2]	**CRC** [5]
VGG19	75.3	80.1	81.3	81.1
BOW-SIFT	25.8	28.5	26.9	27.4
Algorithms	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
VGG19	81.7	83.3	85.6	**86.0**
BOW-SIFT	27.9	29.6	30.5	**31.1**

Table 6: Recognition accuracy (%) of the comparison classifiers on the **Caltech-256** dataset [37]. The results are averaged on 10 independent runs.

4.8. Results on Fine-Grained Visual Recognition

For the **Caltech-UCSD Birds (CUB200-2011)** dataset [38], we adopt the training-test split suggested in [38, 3], and select nearly half of the images in this dataset as the training set and the other half as the test set. On this dataset, as in **Caltech-256**, we also employ two different types of features, i.e., the BoW-SIFT features extracted from [47] and the VGG19 features extracted from VGG-verydeep-19 [49]. The results on recognition accuracy (%) are summarized in Table 7. We observe that the proposed NSCR classifier achieves better performance than the other competing methods. The parameters α, β in our NSCR on this dataset are set as $\alpha = 0.1$ and $\beta = 0.01$.

Algorithms	**Softmax**	**M-SVM** [42]	**SRC** [2]	**CRC** [5]
VGG19	72.1	75.4	76.0	76.2
BOW-SIFT	8.2	10.2	7.7	9.4
Algorithms	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
VGG19	76.2	78.3	79.0	**79.5**
BOW-SIFT	9.1	9.9	10.2	**10.8**

Table 7: Recognition accuracy (%) of the comparison classifiers on the **Caltech-UCSD Birds (CUB200-2011)** dataset [38].

For the **Oxford 102 Flowers** dataset [39], we employ the BoW-SIFT features extracted from [47] and the VGG19 features extracted from VGG-verydeep-19 [49]. The recognition accuracy (%) results are summarized in Table 8. We observe that the proposed NSCR classifier achieves better performance when compared to the other competing methods. The parameters α, β in our NSCR on this dataset are set as $\alpha = 0.01$ and $\beta = 0.1$.

Algorithms	**Softmax**	**M-SVM** [42]	**SRC** [2]	**CRC** [5]
VGG19	87.3	90.9	93.2	93.0
BOW-SIFT	46.5	50.1	47.2	49.9
Algorithms	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
VGG19	93.1	94.8	95.3	**95.7**
BOW-SIFT	49.4	51.2	54.3	**55.1**

Table 8: Recognition accuracy (%) of the comparison classifiers on the **Oxford 102 Flowers** dataset [39].

For the **Aircraft** dataset [40], we also compare with the methods of Symbiotic [43], FV-FGC [28], B-CNN [44]. The features are extracted via a VGG-16 network in an end-to-end manner, as suggested by [3]. The recognition accuracy (%) results are summarized in Table 9. We observe that the proposed NSCR classifier achieves higher accuracy than NRC and B-CNN. This demonstrates that the proposed NSCR classifier can outperforms not only the traditional representation based classifiers such as SRC [2], CRC [5], and NRC [4], but also the CNN based methods like [44] that fine-tunes the pre-trained network on this dataset in an end-to-end manner. The parameters α, β in our NSCR on this dataset are set as $\alpha = \beta = 0.05$.

Algorithms	**Softmax**	**SRC** [2]	**CRC** [5]	**CROC** [6]	**ProCRC** [3]
Acc.	85.7	86.1	86.7	86.9	86.8
Algorithms	**Symbiotic** [43]	**FV-FGC** [28]	**B-CNN** [44]	**NRC** [4]	**NSCR**
Acc.	72.5	80.7	84.1	87.6	**88.3**

Table 9: Recognition accuracy (%) of the comparison classifiers on **Aircraft** dataset [40].

For the **Cars** dataset [41], we use the same training-test split as suggested by [41]: 8,144 images are employed as the training set and the other 8,041 images are employed as the test set. We also compare with Symbiotic [43], FV-FGC [28], B-CNN [44]. The features are extracted via a VGG-16 network [3] in an end-to-end manner. The results on recognition accuracy

(%) are summarized in Table 10. We observe that the proposed NSCR classifier achieves higher accuracy than ProCRC, and the state-of-the-art B-CNN [44] (with the same features). This again demonstrates that the proposed NSCR classifier can outperform both the traditional representation based classifier and the CNN based manner. The parameters α, β in our NSCR on this dataset are set as $\alpha = 0.05$ and $\beta = 0.01$.

Algorithms	**Softmax**	**SRC** [2]	**CRC** [5]	**CROC** [6]	**ProCRC** [3]
Acc.	88.7	89.2	90.0	90.3	90.1
Algorithms	**Symbiotic** [43]	**FV-FGC** [28]	**B-CNN** [44]	**NRC** [4]	**NSCR**
Acc.	78.0	82.7	90.6	90.8	**91.1**

Table 10: Recognition accuracy (%) of the comparison classifiers on the **Cars** dataset [41].

4.9. Comparison on Speed

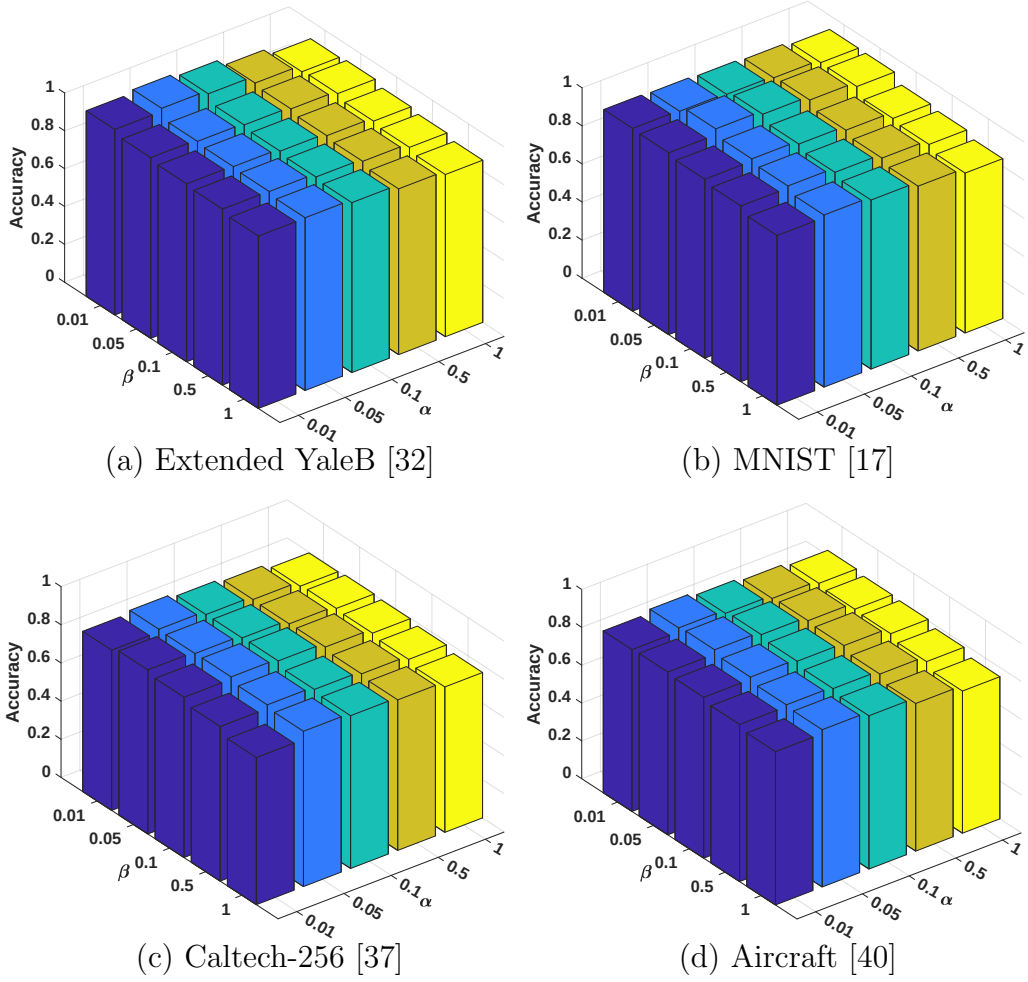
We compare the averaged running time (in seconds) of the proposed NSCR approach and the competing sparse, collaborative, and non-negative representation based classifiers mentioned above, by processing the test images on the **Caltech-256** dataset [37] (we employ 70 images of each class with VGG-19 features as the training set). All experiments are performed with the Matlab2018 environment and run on a computer with 3.50 GHz CPU and 32 GB RAM. In Table 11, we summarize the running time of the comparison methods. Since ProCRC and CRC have closed-form solutions, they are faster than CROC and much faster than iteratively solved SRC. The proposed NSCR classifier also needs $T = 20$ iterations in the ADMM algorithm. It is slower than CRC and NRC, but still faster than SRC.

Algorithms	**Softmax**	**M-SVM** [42]	**SRC** [2]	**CRC** [5]
Time (s)	**0.01**	0.02	0.29	0.05
Algorithms	**CROC** [6]	**ProCRC** [3]	**NRC** [4]	**NSCR**
Time (s)	0.11	0.05	0.10	0.26

Table 11: Running time (in seconds) of the comparison classifiers on the **Caltech-256** dataset [37].

4.10. Parameter Analysis on α and β

Here, we analyze the influence of parameters α and β on the recognition accuracy of the proposed NSCR on four benchmark datasets, e.g., Extended



(a) Extended YaleB [32] (b) MNIST [17]

(c) Caltech-256 [37] (d) Aircraft [40]

Figure 3: Recognition accuracy (%) of NSCR with different parameters α, β on four different benchmark datasets.

YaleB [32] ($d = 300$), MNIST [17] ($N = 500$), Caltech-256 [37] (using VGG19 features), and Aircraft [40]. The results on these datasets are illustrated in Figure 3. We observe that the proposed NSCR model is very robust on the parameters of α, β.

5. Conclusion

In this work, we proposed a jointly non-negative, sparse and collaborative representation (NSCR) to tackle the image recognition problems. The

proposed NSCR can simultaneously extract sparse while discriminative representation for interpretable recognition, and is physically more meaningful than previous sparse, collaborative, or non-negative representation schemes, from the perspective of generative and discriminative property guaranteed by the elastic net [15]. The proposed NSCR representation model is solved by a standard ADMM algorithm, and has closed-form solution in each subproblem. Based on the proposed NSCR representation model, we developed a NSCR based classifier for image recognition. Extensive experiments demonstrate that the proposed NSCR classifier is very efficient and outperforms previous sparse, collaborative, and non-negative representation based classifiers on on diverse challenging image recognition datasets.

Acknowledgement. This work was supported in part by the Major Project for New Generation of AI under Grant 2018AAA0100400, in part by the Natural Science Foundation of China under Grant 61572264, Grant 61620106008, Grant 61802324, and Grant 61772443, in part by the Tianjin Natural Science Foundation under Grant 17JCJQJC43700 and Grant 18ZXZNGX00110, and in part by the Shenzhen Science and Technology Project under Grant GGFW2017040714161462 and JCYJ20170817161916238.

6. Reference

[1] L. J. Li and F. F. Li. What, where and who? classifying events by scene and object recognition. *International Conference on Computer Vision*, pages 1–8, 2007.

[2] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210–227, 2009.

[3] S. Cai, L. Zhang, W. Zuo, and X. Feng. A probabilistic collaborative representation based approach for pattern classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2950–2959, 2016.

[4] J. Xu, W. An, L. Zhang, and D. Zhang. Sparse, collaborative, or nonnegative representation: Which helps pattern classification? *Pattern Recognition*, 88:679–688, 2019.

[5] L. Zhang, M. Yang, and X. Feng. Sparse representation or collaborative representation: Which helps face recognition? *IEEE international conference on Computer vision (ICCV)*, pages 471–478, 2011.

[6] Y. Chi and F. Porikli. Classification and boosting with multiple collaborative representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1519–1531, 2014.

[7] V. N. Vapnik and V. Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.

[8] K.-C. Lee, J. Ho, and D. J. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):684–698, 2005.

[9] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2169–2178. IEEE, 2006.

[10] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 1794–1801. IEEE, 2009.

[11] Q. Zhang and B. Li. Discriminative k-svd for dictionary learning in face recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 2691–2698. IEEE, 2010.

[12] M. Yang, L. Zhang, X. Feng, and D. Zhang. Fisher discrimination dictionary learning for sparse representation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 543–550. IEEE, 2011.

[13] Z. Jiang, Z. Lin, and L. S. Davis. Label consistent k-svd: Learning a discriminative dictionary for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2651–2664, 2013.

[14] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[15] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[16] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[18] J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886, 2013.

[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.

[20] T Cover and P Hart. Nearest neighbor pattern classification. *IEEE Transaction on Information Theory*, 13(1):21–27, 1953.

[21] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13(6):556–562, 2000.

[22] J. Yang, C. Liu, and L. Zhang. Color space normalization: Enhancing the discriminating power of color spaces for face recognition. *Pattern Recognition*, 43(4):1454 – 1466, 2010.

[23] J. Yang, L. Zhang, Y. Xu, and J. Yang. Beyond sparsity: The role of ℓ_1-optimizer in pattern classification. *Pattern Recognition*, 45(3):1104 – 1118, 2012.

[24] M. Yang, Z. Feng, C.K. Shiu, and L. Zhang. Fast and robust face recognition via coding residual map learning based adaptive masking. *Pattern Recognition*, 47(2):535 – 543, 2014.

[25] J. Xie, L. Zhang, J. You, and S. Shiu. Effective texture classification by texton encoding induced statistical features. *Pattern Recognition*, 48(2):447–457, 2015.

[26] Z. Feng, M. Yang, L. Zhang, Y. Liu, and D. Zhang. Joint discriminative dimensionality reduction and dictionary learning for face recognition. *Pattern Recognition*, 46(8):2134 – 2143, 2013.

[27] G. Lin, M. Yang, J. Yang, L. Shen, and W. Xie. Robust, discriminative and comprehensive dictionary learning for face recognition. *Pattern Recognition*, 81:341 – 356, 2018.

[28] P. Gosselin, N. Murray, H. Jégou, and F. Perronnin. Revisiting the fisher vector for fine-grained classification. *Pattern Recognition Letters*, 49:92–98, 2014.

[29] M. Slawski and M. Hein. Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization. *Electron. J. Statist.*, 7:3004–3056, 2013.

[30] L. Breiman. Better subset regression using the nonnegative garrote. *Technometrics*, 37(4):373–384, 1995.

[31] J. Eckstein and D. P. Bertsekas. On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1):293–318, 1992.

[32] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001.

[33] K. Riedel. A sherman-morrison-woodbury identity for rank augmenting matrices with application to centering. *SIAM Journal on Matrix Analysis and Applications*, 13(2):659–662, 1992.

[34] A. M. Martinez and R. Benavente. The ar face database. *CVC Technical Report No. 24*, 1998.

[35] J. J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994.

[36] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. Guibas, and F.-F. Li. Human action recognition by learning bases of action attributes and parts. In *IEEE International Conference on Computer Vision*, pages 1331–1338. IEEE, 2011.

[37] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.

[38] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[39] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008.

[40] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[41] J. Krause, J. Stark, M.and Deng, and F.-F. Li. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[42] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9(Aug):1871–1874, 2008.

[43] Y. Chai, V. Lempitsky, and A. Zisserman. Symbiotic segmentation and part localization for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 321–328, 2013.

[44] T.-Y. Lin, A. RoyChowdhury, and S. Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1449–1457, 2015.

[45] J. Xu, K. Xu, K. Chen, and J. Ruan. Reweighted sparse subspace clustering. *Computer Vision and Image Understanding*, 138(0):25–37, 2015.

[46] J. Xu, M. Yu, L. Shao, W. Zuo, D. Meng, L. Zhang, and D. Zhang. Scaled simplex representation for subspace clustering. *IEEE Transactions on Cybernetics*, pages 1–13, 2019.

[47] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[48] A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. In *Proceedings of the 18th ACM International Conference on Multimedia*, pages 1469–1472. ACM, 2010.

[49] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2014.