Non-negative Sparse and Collaborative Representation for Pattern Classification

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

[a]*Media Computing Lab, 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 pattern classification tasks such as face recognition. In this paper, we propose a novel Non-negative Sparse and Collaborative Representation (NSCR) for pattern classification. The NSCR representation of each test sample is obtained by seeking a non-negative sparse and collaborative representation vector that represents the test sample as a linear combination of training samples. We observe that the non-negativity can make the SR and CR more discriminative and effective for pattern classification. Based on the proposed NSCR, we propose a NSCR based classifier for pattern classification. Extensive experiments on benchmark datasets demonstrate that the proposed NSCR based classifier outperforms the previous SR or CR based approach, as well as state-of-the-art deep approaches, on diverse challenging pattern classification tasks.

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

1. Introduction

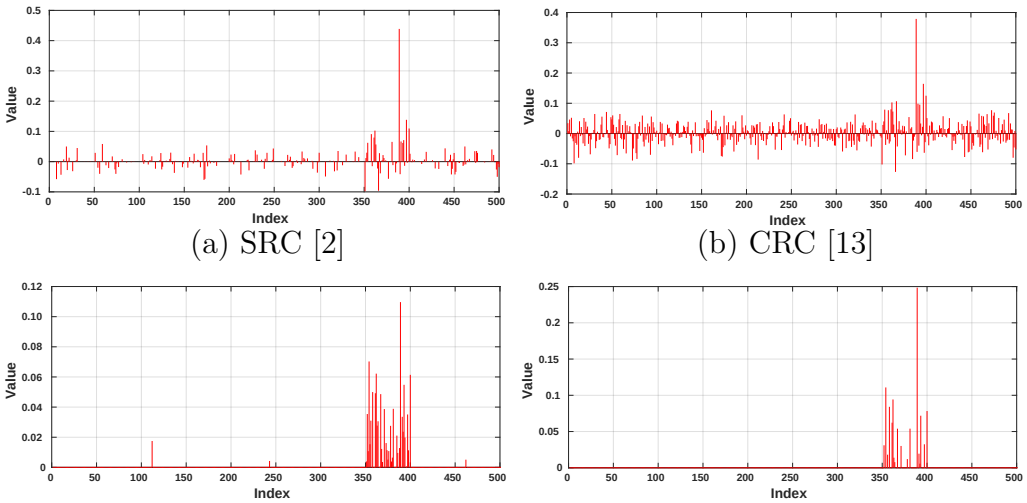
Pattern classification is a supervised learning problem in which the goal is to assign the test images to correct classes according to the labeled training

images. It is a challenging computer vision and image understanding problem with numerous applications [1, 2, 3, 4, 5]. 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 training data vector. $\boldsymbol{X}_k \in \mathbb{R}^{D \times N_k}$ $(k = 1, ..., K)$ is the data matrix of class k, and N_k is the number of training images in this class, the pattern classification task aims to find the correct class that each query data $\boldsymbol{y} \in \mathbb{R}^D$ belongs to.

Numerous pattern classification approaches [2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22] have been proposed for face recognition, action recognition, and object recognition, etc. Among these approaches, one important category is the representation based methods [6, 7, 2, 15, 13, 17, 3, 22]. These approaches first encode a query data $\boldsymbol{y}$ as a linear combination of all the training data $\boldsymbol{X}$, and then assign the query data $\boldsymbol{y}$ to the corresponding class with the minimum reconstruction error on $\boldsymbol{X}_k$ $(k = 1, ..., K)$. Two seminal works in this category are Sparse Representation (SR) based Classifiers (SRC) [2] and Collaborative Representation (CR) based Classifier (CRC) [13], which employ sparse or collaborative properties on the coding coefficients of the query data over the training data, respectively.

Despite the success of SR/CR schemes on the face recognition task, both SR and CR schemes could not avoid to produce bias in the encoding coefficient vectors when the whole training data matrix $\boldsymbol{X}$ are employed to reconstruct the query data $\boldsymbol{y}$. This is because that, from the generative perspective, it is "physically absurd" (but mathematically feasible) to reconstruct a face image by allowing the other images to "cancel each other out" with complex additions and subtractions [23]. Besides, whether it is indeed the sparse or collaborative property that make the SRC/CRC approaches effective is also questioned [24, 25, 13, 22]. To obtain physically more reasonable representation, the Non-negative Representation (NR) based Classifier (NRC) [22] is proposed. However, due to lack of regularization, NRC is not flexible to deal with diverse real-world problems.

In this paper, we propose a novel representation framework by jointly integrating the non-negativity, sparsity and collaborativity for pattern classification. The proposed non-negative sparse and collaborative representation (NSCR) is physically more meaningful and discriminative over previous sparse representation [2, 16], collaborative representation [13, 3], and non-negative representation [22]. In fact, the joint sparse and collaborative representation already enjoys the flexibility as the famous elastic net [26] that linearly combines the ℓ_1 and ℓ_2 penalties of the lasso [27] and ridge regres-



(a) SRC [2] (b) CRC [13]

(c) NRC [22] (d) NSCR

Figure 1: An illustrative comparison of the coding vectors obtained by SRC [2], CRC [13], NRC [22], and the proposed NSCR schemes. The indices 1∼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∼150), $\boldsymbol{X}_5$ (indices 201∼250), and $\boldsymbol{X}_0$ (indices 451∼500). The proposed NSCR produces globally sparse, locally dense, and physically more reasonable representation than other schemes.

sion methods, and its discriminative property for representation based classification can be further boosted by the additional non-negative property we introduced in this work. 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 endmembers in hyperspectral images, just to name a few.

We illustrate the mechanism of the proposed NSCR scheme through an example in Fig. 1. Here, we first select 50 images for each of the 10 handwritten digit numbers $0 \sim 9$ from the MNIST dataset [28]. 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 [29] (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 [13] (Fig. 1 (b)), NRC [22] (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.

Inspired by the advantages of the proposed NSCR representation scheme, we propose a simple yet effective NSCR based classifier for pattern classification. 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 [30]. 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 pattern classification tasks. The contribution of this paper are threefold:

- We propose a novel non-negative sparse and collaborative representation (NSCR) framework as an integration of the sparse, collaborative, and non-negative representation for pattern classification problems.
- Based on the proposed NSCR, we develop a simple yet effective NSCR based classifier for pattern classification.
- 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 classification, etc.

The rest of this paper is 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 pattern classification datasets. In §5, we conclude this paper.

2. Related Work on Representation based Classifiers

Nearest Neighbor based Classifier (NNC) [31] computes independently the distance between the query data and each of the training data, and assigns the label of the training data nearest to the query data as its predicted label. To reduce the computational cost of NNC, nearest subspace classifier (NSC) [7] computes the distance between the query data and each of the subspace and then assigns the query data to its nearest subspace.

Sparse Representation based Classifier(SRC) [2], as a large improvement to NSC, represents the query data as a linear combination of all the training data with ℓ_1 norm sparsity constraint imposed on the representational coefficients, and has shown promising recognition performance ever since its emergence. However, solving an ℓ_1 norm minimization problem would be very slow for processing large-scale datasets.

Collaborative Representation based Classifier (CRC) [13] has been proposed to discuss the role of sparsity or collaborativity on face recognition. CRC represents the query data over all the training data collaboratively with an ℓ_2 norm constraint imposed on the representational coefficients, and demonstrates similar classification accuracy while much faster speed than SRC. Inspired by the success of NSC and CRC, the collaborative representation optimized classifier (CROC) [17] 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 query data belongs to the collaborative subspace of all classes.

Non-negative Representation based Classifier (NRC) [22] is also developed to investigate the use of non-negative representation (NR) for pattern classification. The NR is inspired from the non-negative matrix factorization techniques [23, 32], 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 samples while limiting the representational power of heterogeneous samples, making the representation sparse and discriminative simultaneously and thus providing an effective solution for representation based classifier than SRC/CRC.

Formulation of SRC, CRC, and NRC. SRC [2], CRC [13], NRC [22] and their extensions [3, 17, 33, 34, 35, 36, 37, 38, 39, 40, 41] have been widely studied for various pattern classification tasks such as face recognition, handwritten digit recognition, object recognition, and action recognition, etc. Assume that we have K classes of samples, denoted by $\{\boldsymbol{X}_k\}, k = 1, ..., K$, where $\boldsymbol{X}_k \in \mathbb{R}^{D\times N_k}$ is the sample matrix of class k. Each column of the matrix $\boldsymbol{X}_k$ is a training sample from the k-th class. The whole training sample 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 query sample $\boldsymbol{y} \in \mathbb{R}^D$, both SRC and CRC compute the coding vector $\boldsymbol{c}$ of $\boldsymbol{y}$ over $\boldsymbol{X}$ by solving the following minimization problem:

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

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

where $p = 0$ or 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 sample sub-matrix $\boldsymbol{X}_k$. Assume that the query sample $\boldsymbol{y}$ 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 $\boldsymbol{y}$, 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 samples are most similar to the query sample $\boldsymbol{y}$. The overall classification framework of the SRC/CRC/NRC classifiers is summarized in Algorithm 1.

In this work, we introduce a non-negative sparse and collaborative representation (NSCR) as an integration of the widely studied SR/CR/NR schemes for pattern classification.

3. Non-negative Sparse and Collaborative Representation based Classifier

In this section, we introduce the proposed Non-negative Sparse and Collaborative (NSCR) model, and develop a NSCR representation based classifier for pattern classification.

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

Given a query data $\boldsymbol{y}$ and the training data 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 $\boldsymbol{y}$ over $\boldsymbol{X}$. Here, we integrate the regularization terms of SRC [2] and CRC [13] 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 [26]. As we will see in the next section, the additional non-negative constraint in the proposed NSCR model still make it as efficient as the previous SRC. Since $\boldsymbol{c} > 0$, we have $\|\boldsymbol{c}\|_1 = \boldsymbol{c}$ and Eqn. (4) is equivalent to

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

The proposed NSCR model (5) only contains two explicit parameters (α, β), which makes it especially easy to solve (as will be demonstrated in §3.2). The proposed NSCR can be viewed implicitly as a non-negative constrained elastic LASSO model [26], which provides an intuitive explanation for the experimental findings that it achieves better performance than all previous classifiers such as SRC [2], CR based ones [13, 17, 3], and NRC [22] (please refer to §4 for more details).

There are several advantages in introducing non-negativity into the joint sparse and collaborative representation. Firstly, non-negativity can brings better discriminative ability over SR/CR with higher classification accuracy. This point is validated by the non-negative representation based classifier [22] that, a non-negative representation of the query data could be positive over data from the homogeneous class while zero over data from heterogeneous classes. In SR/CR based classifiers [2, 13, 3], the query data can be approximated by linear combinations of the whole training data. Though this is

mathematically feasible, it is "physically problematic" to reconstruct a sample with complex additions and subtractions in real-world applications [23]. Second, 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 [23]. The non-negativity property allows only non-negative combination of multiple training data to additively reconstruct the query data, which is also compatible with the intuitive notion of combining parts into a whole [23]. Third, non-negativity helps automatically extract sparse and interpretable data representation to reconstruct the query data [42, 43, 44]. For example, the authors in [43] compare the non-negative least square (NNLS) model with the non-negative LASSO [27] and find that the NNLS model can achieve similar or even better performance than non-negative LASSO on sparse recovery.

Although the non-negative constraint is somewhat similar to non-negative matrix factorization (NMF) problem [23, 32], the proposed NSCR model is very different from NMF models. First, we only constraint the coefficients to be non-negative but allow the training data matrix to be arbitrary sample matrix (can contain negative elements). Second, NMF models are often solved via two subproblems, both are non-negative least square problems solved by Alternating Non-Negative Least Squares (ANLS) algiorithms [45]. But our NSCR model only has non-negative constraint on the coefficients, and hence can be solved by standard ADMM algorithms.

3.2. Optimization

Since the proposed NSCR model (5) does not have an analytical solution, we employ the variable splitting method [46, 47] to solve it. We firstly reformulate the NSCR model (5) into a linear equality-constrained problem by introducing an auxiliary variable $\boldsymbol{z}$:

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

Then, the alternating direction method of multipliers(ADMM) [30] framework can be employed to solve the NSCR model (5).

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

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

Algorithm 1: Solve NSCR (6) via ADMM

Input: Query image $\boldsymbol{y}$, training data 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 $\boldsymbol{c}_T$.

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 $\boldsymbol{\delta}$**:

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

This is an elastic net [26] problem without explicit ℓ_1 regularization, and hence can be solved with a closed-form solution:

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

The Eqn. (9) is slow when the number of data N in $\boldsymbol{X}$ is much larger than its 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 [48] 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 precompute 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" operator operates element-wisely.

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

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

The above alternative updatings are repeated until the convergence condition is satisfied or the number of iterations exceeds a preset threshold T. The convergence condition of the ADMM algorithm is: $\|\boldsymbol{z}_{t+1} - \boldsymbol{c}_{t+1}\|_2 \leq \text{Tol}$, $\|\boldsymbol{c}_{t+1} - \boldsymbol{c}_t\|_2 \leq \text{Tol}$, and $\|\boldsymbol{z}_{t+1} - \boldsymbol{z}_t\|_2 \leq \text{Tol}$ are simultaneously satisfied, where $\text{Tol} > 0$ is a small tolerance value. Since the objective function and constraints of NSCR (6) are all strictly convex, the problem (6) solved by the ADMM algorithm is guaranteed to converge to a global optimal solution. These procedures are summarized in Algorithm 1.

Convergence Analysis. The convergence of Algorithm 1 can be guaranteed since the overall objective function (13) is convex with a global optimal solution. 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 approach to 0 simultaneously in 100 iterations. For speed consideration, we set the maximum iteration number as $T = 20$.

3.3. The NSCR based Classifier

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

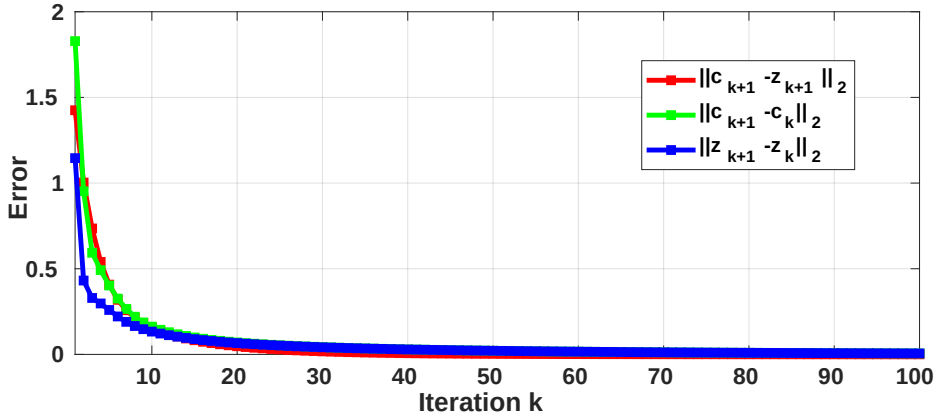


Figure 2: The convergence curves of $\|\mathbf{c}_{t+1} - \mathbf{z}_{t+1}\|_2$ (red line), $\|\mathbf{z}_{t+1} - \mathbf{z}_t\|_2$ (blue line), and $\|\mathbf{c}_{t+1} - \mathbf{c}_t\|_2$ (green line) of the proposed NSCR model (6) on Extended Yale B [49].

Algorithm 2: The NSCR based Classifier

Input: Query sample $\boldsymbol{y}$, 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$(\boldsymbol{y}) = \arg\min_k\{r_k\}$.

3.4. Complexity Analysis

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 [48]. 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 $\mathcal{O}(DN^2T)$, where T is the number of iterations. In Algorithm 2, since the costs of computing the residuals can be ignored, so that the overall cost of Algorithm 2 is $\mathcal{O}(DN^2T)$.

4. Experiments

In this section, we comprehensively compare the proposed Non-negative Sparse and Collaborative Representation (NSCR) based Classifier with state-of-the-art classifiers on diverse pattern classification problems.

4.1. Implementation Details

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

4.2. Datasets

We first introduce two commonly tested datasets for face recognition, i.e., **AR** face dataset [50] and **Extend Yale B** dataset [49]. **AR** face dataset [50] contains over 4,000 color images corresponding to 126 people's faces (70 men and 56 women). Images feature frontal view faces with different facial expressions, illumination conditions, and occlusions (sun glasses and scarf). The pictures were taken under strictly controlled conditions. There is no restriction on wear (clothes, glasses, etc.), make-up, hair style, etc. imposed to participants. The same pictures of each person participated were taken in two sessions, separated by two weeks. **Extend Yale B** dataset [49] has totally 2,432 face images, which are from 38 human subjects, each subject has 64 near frontal images (gray-scale) taken under different illumination conditions. The original images are of 192×168 pixels, and we resize them to 54×48 pixels in our experiments for dimension reduction purposes.

We then introduce two commonly tested datasets for handwritten digit recognition, i.e., **USPS** dataset [51] and **MNIST** dataset [28]. **USPS** dataset [51] contains 9,298 images for handwritten digit numbers from 0 to 9. The training set contains 7,291 images while the testing set contains 2,007 images. Each image is of 16×16 grayscale pixels. **MNIST** dataset [28] of handwritten digits contains 60,000 training images and 10,000 testing images for digit numbers from 0 to 9. The digit in each image has been size-normalized and centered in a fix-size image of 28×28 grayscale pixels.

We also introduce two widely tested datasets for human action recognition and object recognition, respectively, i.e., **Stanford 40 Actions** dataset [52] and **Caltech-256** dataset [53]. **Stanford 40 Actions** dataset [52] 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. **Caltech-256** dataset [53] 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.

We finally introduce four widely tested benchmarks for fine-grained visual classification, i.e., **Caltech-UCSD Birds (CUB200-2011)** dataset [54], **Oxford 102 Flowers** dataset [55], **Aircraft** dataset [56], and **Cars** dataset [57]. **Caltech-UCSD Birds (CUB200-2011)** dataset [54] contains 11,788 bird images. There are totally 200 bird species and the number of images per specie is around 60. The significant variations in pose, viewpoint, and illumination inside each class make this dataset very challenging for visual recognition. **Oxford 102 Flowers** dataset [55] contains 8,189 images from 102 flower classes. Each class has at least 40 images. The flowers appear at different scales, pose, and lighting conditions. This dataset is very challenging since there exist large variations within the same class but small differences across different classes. **Aircraft** dataset [56] contains images of 100 different aircraft model variants, and there are 100 images for each model. The aircrafts appear at different scales, design structures, and appearances, making this dataset very difficult for visual recognition task. **Cars** dataset [57] is consisted of 16,185 images from 196 car classes. Each class has around 80 images with different sizes and heavy clutter background, which make the dataset challenging for visual recognition.

4.3. Comparison Methods

We compare the proposed NSCR with state-of-the-art representation based classifiers such as SRC [2], CRC [13], CROC [17], ProCRC [3], and NRC [22], as well as a widely used discriminative classifier, i.e., linear Support Vector Machine (SVM) [58]. In §4.4, we compare these competing methods on face recognition by using the AR [50] and Extended Yale §4.5, we compare these competing methods on digit recognition by using the USPS [51] and MNIST [28] datasets. We also compare the competing methods on action recognition by using Stanford 40 Actions dataset [52] in §4.6, on object recognition by using Caltech 256 dataset [53] in §4.7, and on several challenging fine-grained visual classification (FGVC) tasks by using Caltech-UCSD Birds-200-2011 [54], Oxford 102 Flowers [55], Aircraft [56], and Cars [57] datasets in §4.8. On FGVC tasks, we also compare with some state-of-art method such as Symbiotic [59], FV-FGC [40], and B-CNN [60].

4.4. Face Recognition

For **AR** dataset [50], as in [2, 13], we choose a subset with only illumination and expression changes that contains 50 male subjects and 50 female subjects from the AR dataset [50] in our experiments. For each subject, the 7 images from Session 1 were used for training, with the other 7 images from Session 2 were used for testing. The images are cropped to 60×43 and normalized to have unit ℓ_2 norm. We project the images to dimension $d = 54, 120, 300$ for further acceleration. The results on classification accuracy (%) of the competing methods are listed in Table 1. We observe that the proposed NSCR approach achieves the highest accuracy results on all cases of projected dimensions, i.e., when d is 54, 120, or 300. The parameters α, β on this dataset are set as $\alpha = \beta = 0.01$.

d	**SVM** [58]	**SRC** [2]	**CRC** [13]	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**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: Classification accuracy (%) of different classifiers on the **AR** dataset [50]. The projected dimension by PCA is d.

For **Extended Yale B** dataset, the original images are of 192×168 pixels. We resize the images to 54×48 pixels and normalize the images to have unit ℓ_2 norm. The images are projected onto a d-dimensional subspace by PCA. Just as the experimental settings in [13], we randomly split the dataset into two halves. Each half contains 32 images for each person. We use one half as the training samples, and the other half as the testing samples. We project the images to dimension $d = 84, 150$, or 300 for further acceleration. The results on classification accuracy (%) of the comparison methods are listed in Table 2. We observe that the proposed NSCR approach achieve better performance than all compared methods on all cases of $d = 84, 120, 300$. The parameters α, β on this dataset are set as $\alpha = 0.05$ and $\beta = 0.01$.

d	**SVM** [58]	**SRC** [2]	**CRC** [13]	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**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: Classification accuracy (%) of different classifiers on the **Extended Yale B** dataset [49]. The results are averaged on 10 independent trials. The projected dimension by PCA is d.

4.5. Handwritten Digit Recognition

For **USPS** dataset [51], similar to the [3], we use all the images in USPS for experiments, and randomly select 50, 100, 200, and 300 images from each digit class of the training set as the training samples, and use all the samples in the testing set as the testing samples. We repeat the experiments on 10 independent trials and report the averaged results. In Table 3, we list the results of classification accuracy (%) by different methods. One can see that the proposed NSCR approach outperforms all the competing methods on all cases of random selecting $N = 50, 100, 200, 300$ images from each class as training samples. With the increasing of the number of training samples, the classification accuracy of all the comparison methods increases consistently, including the proposed NSCR approach. However, one can see that the ProCRC will not perform better when the number of training samples increases from 200 to 300, while the proposed NSCR approach can still increase a lot from 95.3% to 95.7%. We believe that the proposed NSCR approach can achieve higher accuracy when we further increase the number of training samples. Similar trends can also be found on the other algorithms. The parameters α, β on this dataset are set as $\alpha = 0.01$ and $\beta = 0.05$.

N	**SVM** [58]	**SRC** [2]	**CRC** [13]	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**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: Classification accuracy (%) of different classifiers as a function of the number (N) of selected samples from each class for training on the **USPS** dataset [51]. The results are averaged on 10 independent trials.

For **MNIST** dataset [28], we resize each image into 16×16 pixels just as the authors in [3] did. We randomly selected $N = 50, 100, 300, 500$ samples from each class of the training set to construct the training samples, and use all the samples in the testing set as testing samples. Different from [3], for each image in the MNIST dataset, we compute a feature vector by using the scattering convolution network [29]. The feature vector is a concatenation of coefficients in each layer of the network, and is translation invariant and deformation stable. Each feature vector is of size 3,472. The feature vectors for all images are then projected to dimension 500 using PCA. The subspace clustering techniques are then applied to the projected features. We run the experiments on 10 independent trials and only report the averaged classifi-

cation accuracy results (%). In Table 4, we list the results on classification accuracy (%) by the comparison methods. We observe that the proposed NSCR approach achieves comparable performance and even outperforms the comparison methods, no matter how many images (50, 100, 200, or 300) from each class are chosen as the training samples. With the increasing of the number of training samples, the classification accuracy of all the competing methods increases consistently. Due to the advanced features are used by employing the scattering convolution network [29], the classification accuracy of the comparison methods are consistently higher than the corresponding results reported in [3]. The parameters α, β on this dataset are set as $\alpha = 0.05$ and $\beta = 0.05$.

N	**SVM** [58]	**SRC** [2]	**CRC** [13]	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**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: Classification accuracy (%) of different classifiers as a function of the number (N) of selected samples from each class for training on the **MNIST** dataset [28]. The results are averaged on 10 independent trials. The samples are projected onto a 500-dimensional space by PCA.

4.6. Action Recognition

For **Stanford 40 Actions** dataset [52], similar to the experimental settings in [3], we follow the training-test split settings scheme suggested by the authors of [52], and randomly choose 100 images from each class as the training samples and employ the remaining images as the testing samples. Similar as [3], we employ two different types of features to demonstrate the effectiveness of the proposed NSCR approach. For the first type of features, we employ the VLFeat library [61] to extract the Bag-of-Words feature based on SIFT [62] (refer to BOW-SIFT feature). The size of patch and stride are set as 16×16 and 8 pixels, respectively. The codebook is trained by the k-means algorithm, and the size of feature is 1,024. We use a 2-level spatial pyramid representation [8]. The final feature dimension of each image is 5,120. For the second type of features, we use VGG-verydeep-19 [63] to extract CNN features (refer to VGG19 features). We use the activations of the penultimate layer as local features, which are extracted from 5 scales $\{2^s, s = -1, -0.5, 0, 0.5, 1\}$. We pool all local features together regardless of

scales and locations. The final feature dimension of each image is 4,096 for all datasets. Both BOW-SIFT and VGG19 features are ℓ_2 normalized.

The results on classification accuracy (%) by different methods are listed in Table 5. We observe that the proposed NSCR approach achieves higher accuracy than the previous representation based methods such as SRC [2], CRC [13], and NRC [22], no matter what types of features are employed. This demonstrates that NSCR is indeed more effective than previous SR, CR, and NR based classifiers. The parameters α, β on this dataset are set as $\alpha = 0.05$ and $\beta = 0.1$.

Algorithms	**Softmax**	**M-SVM** [58]	**SRC** [2]	**CRC** [13]
VGG19	77.2	79.0	78.7	78.2
BOW-SIFT	21.1	24.0	24.2	24.6
Algorithms	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**NSCR**
VGG19	79.2	80.9	81.9	**82.3**
BOW-SIFT	24.5	28.4	29.2	**29.5**

Table 5: Classification accuracy (%) of different classifiers on the **Stanford 40 Actions** dataset [52].

4.7. Object Recognition

For **Caltech-256** dataset [53], following the common experimental settings in [3], we randomly choose 15, 30, 45, and 60 images from each object category as the training samples, respectively, and use the remaining images as the testing samples. For fair comparison, we also run the proposed NSCR and the comparison methods for 10 independent trials for each partition and only report the averaged classification accuracy results (%). Similar to the operations on the **Stanford 40 Actions** dataset [52] mentioned above, we employ two different types of features, i.e., the VGG19 features extracted from VGG-verydeep-19 [63] and the BoW-SIFT features extracted from [62]. The results on classification accuracy (%) are listed in Table 6. We observe that the proposed NSCR approach still achieves the best recognition performance when compared to the other competing methods. The parameters α, β on this dataset are set as $\alpha = 0.01$ and $\beta = 0.05$.

4.8. Fine-Grained Visual Classification

For **Caltech-UCSD Birds (CUB200-2011)** dataset [54], we adopt the publicly available split [54, 3], and use nearly half of the images in this

Algorithms	**Softmax**	**M-SVM** [58]	**SRC** [2]	**CRC** [13]
VGG19	75.3	80.1	81.3	81.1
BOW-SIFT	25.8	28.5	26.9	27.4
Algorithms	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**NSCR**
VGG19	81.7	83.3	85.6	**86.0**
BOW-SIFT	27.9	29.6	30.5	**31.1**

Table 6: Classification accuracy (%) of different classifiers on the **Caltech-256** dataset [53]. The results are averaged on 10 independent trials.

dataset as the training samples and the other half as the testing samples. For this dataset, just as the operations on **Caltech-256**, we also employ two different types of features, i.e., the VGG19 features extracted from VGG-verydeep-19 [63] and the BoW-SIFT features extracted from [62]. The results on classification accuracy (%) are listed in Table 7. We observe that the proposed NSCR approach still achieves higher recognition performance when compared to the other competing methods. The parameters α, β on this dataset are set as $\alpha = 0.1$ and $\beta = 0.01$.

Algorithms	**Softmax**	**M-SVM** [58]	**SRC** [2]	**CRC** [13]
VGG19	72.1	75.4	76.0	76.2
BOW-SIFT	8.2	10.2	7.7	9.4
Algorithms	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**NSCR**
VGG19	76.2	78.3	79.0	**79.5**
BOW-SIFT	9.1	9.9	10.2	**10.8**

Table 7: Classification accuracy (%) of different classifiers on the **Caltech-UCSD Birds (CUB200-2011)** dataset [54].

For **Oxford 102 Flowers** dataset [55], just as the operations on **Caltech-256**, we still employ two different types of features, i.e., the VGG19 features extracted from VGG-verydeep-19 [63] and the BoW-SIFT features extracted from [62]. The results on classification accuracy (%) are listed in Table 8, from which we observe that the proposed NSCR approach still achieves higher recognition performance when compared to the other competing methods. The parameters α, β on this dataset are set as $\alpha = 0.01$ and $\beta = 0.1$.

For **Aircraft** dataset [56], we also compare with the Symbiotic [59], FV-FGC [40], B-CNN method [60]. The features are extracted by using a VGG-16 network [3] by end-to-end manner. The results on classification accuracy (%)

Algorithms	**Softmax**	**M-SVM** [58]	**SRC** [2]	**CRC** [13]
VGG19	87.3	90.9	93.2	93.0
BOW-SIFT	46.5	50.1	47.2	49.9
Algorithms	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**NSCR**
VGG19	93.1	94.8	95.3	**95.7**
BOW-SIFT	49.4	51.2	54.3	**55.1**

Table 8: Classification accuracy (%) of different classifiers on the **Oxford 102 Flowers** dataset [55].

are listed in Table 9. We observe that the proposed NSCR approach achieves higher accuracies than those of the NRC and B-CNN methods. These results demonstrate that the proposed NR based classifier can outperforms not only the traditional representation based classifiers such as SRC, CRC, and NRC, but also the CNN based methods that fine-tunes the pre-trained network on this dataset in an end-to-end manner. The parameters α, β on this dataset are set as $\alpha = 0.05$ and $\beta = 0.05$.

Algorithms	**Softmax**	**SRC** [2]	**CRC** [13]	**CROC** [17]	**ProCRC** [3]
Acc.	85.7	86.1	86.7	86.9	86.8
Algorithms	**Symbiotic** [59]	**FV-FGC** [40]	**B-CNN** [60]	**NRC** [22]	**NSCR**
Acc.	72.5	80.7	84.1	87.6	**88.3**

Table 9: Classification accuracy (%) of different classifiers on the **Aircraft** dataset [56].

For **Cars** dataset , We use the same split provided by [57], in which 8,144 images are employed as the training samples and the other 8,041 images are employed as the testing samples. We also compare with the Symbiotic [59], FV-FGC [40], B-CNN [60]. The features are extracted by using a VGG-16 network [3] by end-to-end manner, just the same as the settings on the **Aircraft** dataset. The results on classification accuracy (%) are listed in Table 10. One can see that the proposed NSCR approach achieves 0.7 present higher than ProCRC, while 0.2 percent higher than the state-of-the-art B-CNN method (with the same features) [60]. These results demonstrate that the proposed NR based classifier can outperform both the traditional representation based classifier and the CNN based manner. The parameters α, β on this dataset are set as $\alpha = 0.05$ and $\beta = 0.01$.

Algorithms	**Softmax**	**SRC** [2]	**CRC** [13]	**CROC** [17]	**ProCRC** [3]
Acc.	88.7	89.2	90.0	90.3	90.1
Algorithms	**Symbiotic** [59]	**FV-FGC** [40]	**B-CNN** [60]	**NRC** [22]	**NSCR**
Acc.	78.0	82.7	90.6	90.8	**91.1**

Table 10: Classification accuracy (%) of different classifiers on the **Cars** dataset [57].

4.9. Comparison on Speed

We compare the running time (second) of the proposed NSCR approach and the competing representation based classifiers mentioned above by processing one test image on the **Caltech-256** dataset [53] (70 samples from each class and VGG-19 features for training). All experiments are run under the Matlab environment and run on a machine with 3.50 GHz CPU and 32 GB RAM. Table 11 lists the running time of different methods. The ProCRC and CRC methods have closed-form solutions and have the same speed, which is faster than CROC and much faster than SRC. The proposed NSCR approach need 20 iterations in the ADMM algorithm and hence is slower than CRC and NRC, but still faster than SRC.

Algorithms	**Softmax**	**M-SVM** [58]	**SRC** [2]	**CRC** [13]
Time (s)	**0.01**	0.02	0.29	0.05
Algorithms	**CROC** [17]	**ProCRC** [3]	**NRC** [22]	**NSCR**
Time (s)	0.11	0.05	0.10	0.26

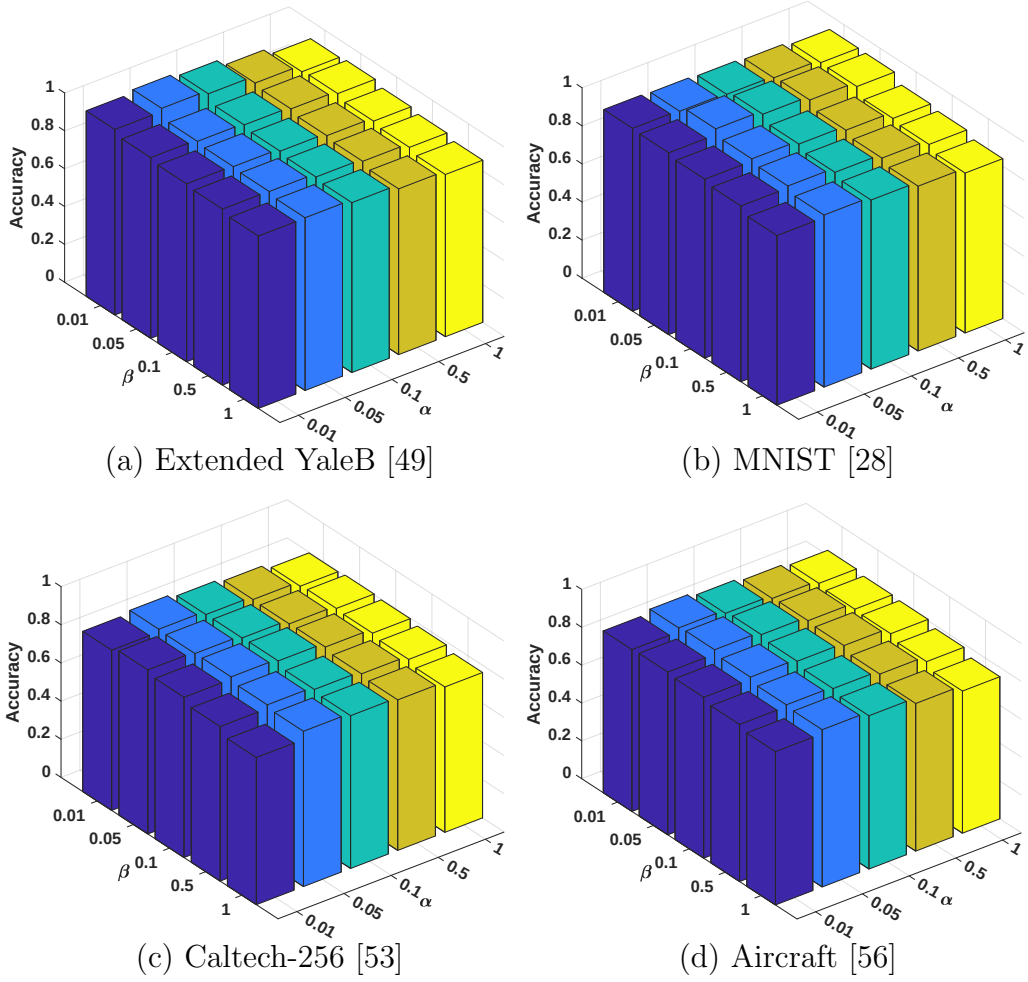
Table 11: Running time (in second) of different methods on the **Caltech-256** dataset [53].

4.10. Parameter Analysis on α and β

Here, we analyze the influence of parameters α and β on the classification accuracy of the proposed NSCR on four benchmark datasets, e.g., Extended YaleB [49] ($d = 300$), MNIST [28] ($N = 500$), Caltech-256 [53] (using VGG19 features), and Aircraft [56]. 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 non-negative sparse and collaborative representation (NSCR) for pattern classification. The proposed NSCR can automatically extract sparse, collaborative and interpretable data for better



(a) Extended YaleB [49] (b) MNIST [28]

(c) Caltech-256 [53] (d) Aircraft [56]

Figure 3: Classification accuracy of NSCR with different parameters α, β on four different benchmark datasets.

classification performance, 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 [26]. The proposed NSCR model is solved by a standard ADMM algorithm, and has closed-form solution in each subproblem. Based on the proposed NSCR scheme, we developed a NSCR based classifier for pattern classification. Extensive experiments on diverse challenging datasets demonstrate that the proposed NSCR based classifier is very efficient and outperforms previous representation based classifiers on pattern classification problems.

Acknowledgement. This work is partially supported by the Shenzhen Science and Technology Project under Grant GGFW2017040714161462 and JCYJ20170817161916238.

6. Reference

[1] L. J. Li and Li Fei-Fei. 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] Sijia Cai, Lei Zhang, Wangmeng Zuo, and Xiangchu 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] Zhou Xu, Shuai Li, Yutian Tang, Xiapu Luo, Tao Zhang, Jin Liu, and Jun Xu. Cross version defect prediction with representative data via sparse subset selection. In *Proceedings of the 26th Conference on Program Comprehension*, pages 132–143. ACM, 2018.

[5] Zhou Xu, Shuai Li, Xiapu Luo, Jin Liu, Tao Zhang, Yutian Tang, Jun Xu, Peipei Yuan, and Jacky Keung. Tstss: A two-stage training subset selection framework for cross version defect prediction. *Journal of Systems and Software*, 154:59–78, 2019.

[6] Vladimir Naumovich Vapnik and Vlamimir Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.

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

[8] 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.

[9] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang. Linear spatial pyramid matching using sparse coding for image classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1794–1801. IEEE, 2009.

[10] S. Gao, I. W. H. Tsang, L. T. Chia, and P. Zhao. Local features are not lonely-laplacian sparse coding for image classification. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3555–3561, 2010.

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

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

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

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012.

[15] Li Zhang, Wei-Da Zhou, Pei-Chann Chang, Jing Liu, Zhe Yan, Ting Wang, and Fan-Zhang Li. Kernel sparse representation-based classifier. *IEEE Transactions on Signal Processing*, 60(4):1684–1695, 2012.

[16] 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.

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

[18] T. H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma. Pcanet: A simple deep learning baseline for image classification? *IEEE Transactions on Image Processing*, 24(12):5017–5032, 2015.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[20] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[21] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[22] Jun Xu, Wangpeng An, Lei Zhang, and David Zhang. Sparse, collaborative, or nonnegative representation: Which helps pattern classification? *Pattern Recognition*, 88:679 – 688, 2019.

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

[24] Hazım Kemal Ekenel and Rainer Stiefelhagen. Why is facial occlusion a challenging problem? In *Advances in Biometrics: Third International Conference*, pages 299–308, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[25] R. Rigamonti, M. A. Brown, and V. Lepetit. Are sparse representations really relevant for image classification? In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1545–1552, 2011.

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

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

[28] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

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

[30] 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.

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

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

[33] 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.

[34] 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.

[35] 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.

[36] 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.

[37] 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.

[38] 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.

[39] R. Timofte and L. Van Gool. Adaptive and weighted collaborative representations for image classification. *Pattern Recognition Letters*, 43:127–135, 2014.

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

[41] Yang Wu, Wei Li, Masayuki Mukunoki, Michihiko Minoh, and Shihong Lao. Discriminative collaborative representation for classification. In *Asian Conference on Computer Vision*, pages 205–221, 2014.

[42] Nicolas Gillis. The why and how of nonnegative matrix factorization. *Regularization, Optimization, Kernels, and Support Vector Machines*, 12(257), 2014.

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

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

[45] Andrzej Cichocki, Rafal Zdunek, and Shun Ichi Amari. Hierarchical als algorithms for nonnegative matrix and 3d tensor factorization. *Springer Lncs*, 4666:169–176, 2007.

[46] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bull. Amer. Math. Soc.*, 49(1):1–23, 1943.

[47] 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.

[48] 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.

[49] 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.

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

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

[52] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1331–1338. IEEE, 2011.

[53] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

[54] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[55] 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.

[56] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[57] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[58] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9(Aug):1871–1874, 2008.

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

[60] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1449–1457, 2015.

[61] Andrea Vedaldi and Brian 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.

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

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