Sparse Subspace Clustering for Incomplete Images

Xiao Wen[1], Linbo Qiao[2], Shiqian Ma[1], Wei Liu[3], and Hong Cheng[1]

[1]Department of SEEM, CUHK, Hong Kong. {wenx, sqma, hcheng}@se.cuhk.edu.hk
[2]College of Computer, National University of Defense Technology, China. qiao.linbo@nudt.edu.cn
[3]Didi Research, Beijing, China. weiliu@didichuxing.com

Abstract

In this paper, we propose a novel approach to cluster incomplete images leveraging sparse subspace structure and total variation regularization. Sparse subspace clustering obtains a sparse representation coefficient matrix for input data points by solving an ℓ_1 minimization problem, and then uses the coefficient matrix to construct a sparse similarity graph over which spectral clustering is performed. However, conventional sparse subspace clustering methods are not exclusively designed to deal with incomplete images. To this end, our goal in this paper is to simultaneously recover incomplete images and cluster them into appropriate clusters. A new nonconvex optimization framework is established to achieve this goal, and an efficient first-order algorithm is developed to tackle the nonconvex optimization. Extensive experiments carried out on three public datasets show that our approach can restore and cluster incomplete images very well when up to 30% image pixels are missing.

1. Introduction

With the emergence of new sensory elements, improvements of computing nodes, and bandwidth growths of wireless communication in intelligent sensing systems, the past few years have witnessed an explosion of high-dimensional data. In a variety of problems such as geographic information systems, spatial databases, solid modeling, computer vision, computational geometry, and bioinformatics, high-dimensional data is ubiquitous. High-dimensional data not only provides opportunities to discover hidden patterns and correlations, but also inspires higher computational costs and memory requirements. However, high-dimensional data is usually, for example in face clustering and motion segmentation, supposed to lie in relatively low-dimensional one or multiple subspaces. In other words, the intrinsic dimension of data is much lower than the original dimension. For example, as illustrated in [23], in handwritten digit recognition the images from a single digit can be well approximated by a 12-dimensional subspace. [20] shows that in motion segmentation the feature points of a single rigid motion will lie in a linear subspace of dimension at most four. Likewise, in face clustering the images under different lighting conditions belonging to one person can be well approximated by a 5-dimensional subspace [19].

Subspace clustering naturally arises with the emergence of high-dimensional data. It refers to the problem of finding multiple low-dimensional subspaces underlying a collection of data points sampled from a high-dimensional space and simultaneously partitioning these data into subspaces. Making use of the low intrinsic dimension of input data and the assumption of data lying in a union of linear or affine subspaces, subspace clustering assigns each data point to a subspace, in which data points residing in the same subspace belong to the same cluster. Subspace clustering has been applied to various areas such as computer vision [10], signal processing [15], and bioinformatics [12]. In the past two decades, numerous algorithms to subspace clustering have been proposed, including K-plane [3], GPCA [24], Spectral Curvature Clustering [4], Low Rank Representation (LRR) [14], Sparse Subspace Clustering [8], *etc*. Among these algorithms, the recent work of Sparse Subspace Clustering (SSC) has been recognized to enjoy promising empirical performance. SSC assumes that each data point can be represented as a linear combination of other points that belong in the same subspace, and then finds a sparse representation coefficient matrix by which a weighted graph is constructed. On the graph, a conventional spectral clustering algorithm can be applied to assign each data point to a cluster, tolerating sparse noise or outliers. When some entries of the data matrix are corrupted or missing, the drawback of SSC is that it simply disregards the fact that we know the locations of the missing entries in the data matrix, though sometimes we may have no idea about the range or distribution

of the corresponding values. Another approach suggested in [8] deletes the rows of the data matrix that involve missing values, and then clusters the data using only the remaining rows. To explain how this approach works, let us consider a data matrix $M = [M_1, \ldots, M_N] \in \mathbb{R}^{d\times N}$, where each column vector $M_i \in \mathbb{R}^d$ represents one data point. Let $J_i \subset \{1, ..., d\}$ denote the indices of the known entries of M_i and define by $J = \bigcap_{i=1}^N J_i$. If we only keep the rows of M that correspond to J, we can obtain a reduced data matrix $\bar{M} \in \mathbb{R}^{|J|\times N}$ with only known entries. Sparse subspace clustering is then performed over the reduced matrix. The drawback of this approach is that the size of J is quite small relative to the ambient space dimension d, and much useful information (*e.g.*, known pixel values) is thus ignored. In fact, for the images considered in this paper, we find that J is always empty when more than 10% entries are missing with their locations randomly chosen, because missing pixels are present in every row of the data matrix M. Recently, [25] showed that a modified version of SSC is provably effective in correctly identifying the underlying subspaces for even noisy data. This extends the theoretical guarantees of SSC to practical settings, and provides a justification to the success of SSC in a class of applications. [18] proposed a greedy algorithm to cluster incomplete data. However, such a greedy algorithm lacks theoretical convergence guarantees and is time consuming. Due to all these limitations, as pointed out in [8], addressing the problem of subspace clustering under a large fraction of missing entries is a challenging topic, which has not been thoroughly explored yet. This challenge is exactly what we will address in this paper. While this work focuses on optimization models for tackling subspace clustering, we note that there exist some work that deals with subspace clustering from a probabilistic perspective (see [9] and the references therein).

In this paper, we propose a novel approach beyond SSC to cluster incomplete images. Compared with the approaches mentioned above, the key advantage of our approach is that it can accomplish image restoration and subspace clustering simultaneously. The model that we build for clustering incomplete images incorporates popular total variation (TV) regularization [21] into image restoration. By integrating SSC and TV together, our model formulates subspace clustering as a structured nonconvex optimization problem. This formulation generalizes traditional SSC, and captures the global structure of incomplete image data. We present a first-order algorithm to solve the nonconvex optimization, and further prove that it converges to a KKT point of the nonconvex problem under certain standard assumptions. Extensive experiments on the Extended Yale B dataset [13], the USPS digital images dataset [11], and the Columbia Object Image Library (COIL20) [16] show that for images with up to 30% missing pixels the clustering quality achieved by our approach compares favorably to the state-of-the-arts.

The rest of the paper is organized as follows. In Section 2, we propose our optimization model that incorporates SSC with TV for incomplete images clustering. We introduce a first-order algorithm, i.e., the so-called alternating direction method of multipliers, to solve our optimization problem in Section 3. In Section 4, we analyze the convergence property of the proposed algorithm. Section 5 shows the numerical results on three data sets: the Extended Yale B dataset, the USPS digital images dataset, and the Columbia Object Image Library. Conclusions are drawn in Section 6.

2. SSC with TV Regularization

2.1. Sparse Subspace Clustering

In this section we introduce the basic idea and formulation of SSC proposed by Elhamifar and Vidal [7, 8]. SSC takes advantage of the so-called *self-expressiveness property*. To be specific, suppose we want to cluster N data points $M := [M_1, \ldots, M_N]$ to L subspaces $\mathcal{S}_1, \ldots, \mathcal{S}_L$. The key idea of SSC is to find a sparse expression of each column M_i as a linear combination of all other columns of M, i.e., $M_i = MC_i$, where $C_i \in \mathbb{R}^N$ is the coefficient vector. It is believed that C_i should be a sparse vector with the i-th entry being zero, as each data point is expected to be represented only by the other data points lie in the same subspace. This motivates the following optimization problem on coefficient matrix $C := [C_1, \ldots, C_N] \in \mathbb{R}^{N\times N}$.

$$\min \ \|C\|_1, \quad \text{s.t.}, M = MC, \text{diag}(C) = 0, \tag{1}$$

where $\|C\|_1 = \sum_{i,j} |C_{i,j}|$, $\text{diag}(C)$ represents the diagonal vector of C. It is suggested in [7] that the affinity matrix W can be defined as $W = |C| + |C|^T$, i.e., $W_{ij} = |C_{ij}| + |C_{ji}|$. A weighted graph $\mathcal{G} := (\mathcal{V}, \mathcal{E}, W)$ is then constructed, where $\mathcal{V}$ denotes the set of N nodes corresponding to the N data points, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges between nodes. Clustering of data into subspaces follows by applying spectral clustering [17] to the graph $\mathcal{G}$.

In real applications, data are often corrupted by noise or sparse outliers. In [8], the following variant of SSC is proposed to deal with data corruption:

$$\begin{array}{ll} \min & \|C\|_1 + \frac{\rho}{2}\|Z\|_F^2 + \nu\|E\|_1, \\ \text{s.t.} & \text{diag}(C) = 0, Z = M - MC - E, \end{array} \tag{2}$$

where Z and E correspond to noise and sparse outliers respectively, $\rho > 0$, $\nu > 0$ are two trade-off parameters balancing the three terms in the objective function.

In our problem, we assume that the data matrix M is incomplete and corrupted only by noise, so E is eliminated from optimization problem (2). In the application of images clustering, incomplete data indicate that there are missing pixels in the images. An SSC-based approach for this in-

complete images clustering problem can be formulated as

$$\begin{array}{ll} \min_{C,Y} & \|C\|_1 + \frac{\rho}{2}\|Y - YC\|_F^2, \\ \text{s.t.}, & \text{diag}(C) = 0, Y_{ij} = M_{ij}, (i,j) \in \Omega, \end{array} \tag{3}$$

where Ω denotes the set of indices of the known pixels of the images M. Solving (3) recovers the images M and obtains the sparse coefficient matrix C simultaneously. However, note that problem (3) ignores the fact that each column of M corresponds to an image and thus adding some regularization term for Y is expected to achieve better recovery quality. Here the total variation regularization is adopted for this purpose.

2.2. SSC with TV Regularization

Removing noise from images using total variation regularization was first proposed in [21]. For a two-dimensional signal $U \in \mathbb{R}^{m\times n}$, such as an image, the total variation function is defined as

$$\text{TV}(U) := \sum_{i,j} \|(U_{i+1,j} - U_{i,j}; U_{i,j+1} - U_{i,j})\|_2 = \sum_{q=1}^{mn} \|D_q u\|_2,$$

where $u := \text{vec}(U) \in \mathbb{R}^{mn}$ is the vector form of matrix U, and D_q denotes the first-order horizontal and vertical finite differences of u at the q-th pixel.

Our approach naturally combines SSC and TV to achieve the goal of images clustering and restoration simultaneously. Our optimization model for SSC with TV regularization (SSC-TV) can be formulated as (assuming that we have N images with size $m \times n$):

$$\begin{array}{ll} \min\limits_{C,Y} & \|C\|_1 + \frac{\rho}{2}\|Y - YC\|_F^2 + \beta \sum\limits_{p=1}^{N}\sum\limits_{q=1}^{mn} \|D_q Y_p\|_2 \\ \text{s.t.} & \text{diag}(C) = 0, Y_{ij} = M_{i,j}, (i,j) \in \Omega, \end{array} \tag{4}$$

where $Y = [Y_1, \ldots, Y_N]$, $Y_p \in \mathbb{R}^{mn}, p = 1, \ldots, N$, and $\rho > 0, \beta > 0$ are two parameters balancing the three terms in the objective function. Note that TV regularization is adopted for each image Y_p.

3. Algorithm

In this section, we propose an efficient first-order algorithm, the so-called alternating direction method of multipliers (ADMM), to solve (4). ADMM has been proved to be very efficient for solving structured convex optimization problems (see [2] for a recent survey on this topic). Here we adopt this algorithm to solve the nonconvex optimization problem (4).

By introducing auxiliary variables $W_{pq} = D_q Y_p$, (4) can be reformulated as

$$\begin{array}{ll} \min & \|C\|_1 + \frac{\rho}{2}\|Y - YC\|_F^2 + \beta \sum\limits_{p=1}^{N}\sum\limits_{q=1}^{mn} \|W_{pq}\|_2 \\ \text{s.t.} & W_{pq} = D_q Y_p, \forall p, q, \quad Y \in S_1, C \in S_2, \end{array} \tag{5}$$

where $S_1 := \{Y \mid Y_{ij} = M_{ij}, (i,j) \in \Omega\}$ and $S_2 := \{C \mid \text{diag}(C) = 0\}$. The augmented Lagrangian function of (5) without considering the two constraints $Y \in S_1$ and $C \in S_2$, is given by

$$\begin{array}{rl} & \mathcal{L}(Y, C, W; E) \\ := & \|C\|_1 + \frac{\rho}{2}\|Y - YC\|_F^2 + \beta \sum_{p=1}^{N}\sum_{q=1}^{mn} \|W_{pq}\|_2 \\ & - \sum_{pq}\langle W_{pq} - D_q Y_p, E_{pq}\rangle + \frac{\gamma}{2}\sum_{pq}\|W_{pq} - D_q Y_p\|_2^2, \end{array}$$

where E_{pq} denotes the Lagrange multiplier associated with the equality constraint $W_{pq} = D_q Y_p$, and $\gamma > 0$ is a penalty parameter.

Given (Y^k, C^k, W^k, E^k), ADMM iterates as follows

$$Y^{k+1} := \operatorname*{argmin}_{Y \in S_1} \mathcal{L}(Y, C^k, W^k; E^k), \tag{6a}$$

$$C^{k+1} := \operatorname*{argmin}_{C \in S_2} \mathcal{L}(Y^{k+1}, C, W^k; E^k), \tag{6b}$$

$$W^{k+1} := \operatorname*{argmin}_{W} \mathcal{L}(Y^{k+1}, C^{k+1}, W; E^k), \tag{6c}$$

$$E_{pq}^{k+1} := E_{pq}^k - \gamma(W_{pq}^{k+1} - D_q Y_p^{k+1}), \forall p, q. \tag{6d}$$

Now we show how to solve the three subproblems in (6a), (6b) and (6c). Note that (6a) reduces to a quadratic programming (QP). Although there are many existing algorithms for solving a QP, we here use a simple linearization strategy to approximate this problem, as it is a subproblem and does not need to be solved exactly all the time. Our linearization strategy for approximating (6a) is to take a gradient projection step for it, i.e., (6a) is replaced by

$$\min_{Y \in S_1} \|Y - (Y^k - \tau_1 g_Y^k)\|_F^2, \tag{7}$$

where $\tau_1 > 0$ is the step size for the gradient step,

$$\begin{array}{rl} g_Y^k := & \rho Y^k (I - C^k)(I - C^k)^T + \sum_{pq} D_q^T E_{pq}^k e_p^T \\ & + \gamma \sum_{pq}(D_q^T D_q Y^k e_p e_p^T - D_q^T W_{pq}^k e_p^T) \end{array}$$

is the gradient of $\mathcal{L}(Y, C^k, W^k; E^k)$ at the current iterate Y^k, and e_p denotes the p-th unit vector. Note that (7) corresponds to an easy projection onto S_1.

Note that (6b) and (6c) can actually be solved simultaneously. To solve (6b), we apply a similar linearization strategy. Specifically, we replace (6b) by

$$\min_{C \in S_2} \|C\|_1 + \frac{1}{2\tau_2}\|C - (C^k - \tau_2 g_C^k)\|_F^2, \tag{8}$$

where $\tau_2 > 0$ is the step size for the gradient step, and $g_C^k := -\rho(Y^{k+1})^T(Y^{k+1} - Y^{k+1}C^k)$ is the gradient of $\frac{\rho}{2}\|Y^{k+1} - Y^{k+1}C\|_F^2$ at the current iterate C^k. Note that in this subproblem we only linearize the quadratic part, because the other part $\|C\|_1$ is nonsmooth. This technique has been widely used in algorithms for solving sparse reconstruction problem arising from compressed sensing (see,

e.g., [1, 28]). Note that without considering the constraint $C \in S_2$, (8) admits an easy closed-form solution known as the soft-shrinkage operation $\bar{C}^{k+1} := \text{shrink}_1(C^k - \tau_2 g_C^k, \tau_2)$, where the soft-shrinkage operator is defined as (see [5])

$$[\text{shrink}_1(Z, \xi)]_{ij} := \text{sign}(Z_{ij}) \bullet \max\{|Z_{ij}| - \xi, 0\}.$$

Now the solution of (8), i.e., C^{k+1} is obtained by setting the diagonal of $\bar{C}^{k+1}$ to zero. (6c) can be reduced to

$$\min_W \frac{\gamma}{2} \sum_{pq} \|W_{pq} - D_q Y_p^{k+1}\|_2^2 - \sum_{pq} \langle W_{pq}, E_{pq}^k \rangle + \beta \sum_{pq} \|W_{pq}\|_2, \tag{9}$$

which admits a closed-form solution given by the so-called ℓ_2-shrinkage operation $W_{pq}^{k+1} := \text{shrink}_2(D_q Y_p^{k+1} + E_{pq}^k/\gamma, \beta/\gamma), \forall p, q$, where the ℓ_2-shrinkage operator is defined as (see [27]) $\text{shrink}_2(z, \xi) := \frac{z}{\|z\|_2} \bullet \max\{\|z\|_2 - \xi, 0\}$, where we adopted the convention $0 \cdot (0/0) = 0$.

Thus we have shown that the three subproblems (6a), (6b) and (6c) in the ADMM algorithm can be (approximately) solved relatively easily. Note that all the three reformulations in (7), (8) and (9) admit simple closed-form solutions. We now summarize our ADMM algorithm for solving (5) in Algorithm 1.

Algorithm 1 ADMM algorithm for solving problem (5)

Input: Y^0, C^0, W^0, E^0
Output: Y, C
1: Update Y^{k+1} by solving (7)
2: Update C^{k+1} by solving (8)
3: Update W^{k+1} by solving (9)
4: Update E^{k+1} by (6d)

4. Convergence Analysis

Similar to [22, 26], in this section, we provide a convergence analysis for the proposed ADMM algorithm showing that under certain standard conditions, any limit point of the iteration sequence generated by Algorithm 1 is a KKT point of (5).

Theorem 1. *Let $X := (Y, C, W, E)$ and $\{X^k\}_{k=1}^\infty$ be generated by Algorithm 1. Assume that $\{X^k\}_{k=1}^\infty$ is bounded and $\lim_{k\to\infty}(X^{k+1} - X^k) = 0$. Then any accumulation point of $\{X^k\}_{k=1}^\infty$ is a KKT point of problem (5).*

Due to the space limit, the proof of Theorem 1 is placed in the supplemental material. It should be pointed out that the assumptions in Theorem 1 are standard in the literature. For example, these assumptions were also used in [22, 26].

5. Experiments

In this section, we apply our SSC-TV approach to three different data sets. As SSC proposed in [8] is considered as one of the state-of-the-art algorithms for subspace clustering, we will mainly focus on the comparison of our approach with SSC [8]. The MATLAB codes of SSC were downloaded from http://www.cis.jhu.edu/∼ehsan/code.htm.

5.1. Post-Processing and Spectral Clustering

After solving (4), we obtain the recovered images Y and the sparse coefficient matrix C. Similar to [6], we perform some post-processing procedure on C. For each coefficient vector C_i, we keep its largest T coefficients in absolute magnitude and set the remaining coefficients to zeros. The affinity matrix W associated with the weighted graph $\mathcal{G}$ is then constructed as $W = |C| + |C|^T$. To obtain the final clustering result, we apply the normalized spectral clustering approach proposed by Ng *et al* [17]. Thus, the whole procedure of our SSC-TV based clustering approach for incomplete images can be described as in Algorithm 2.

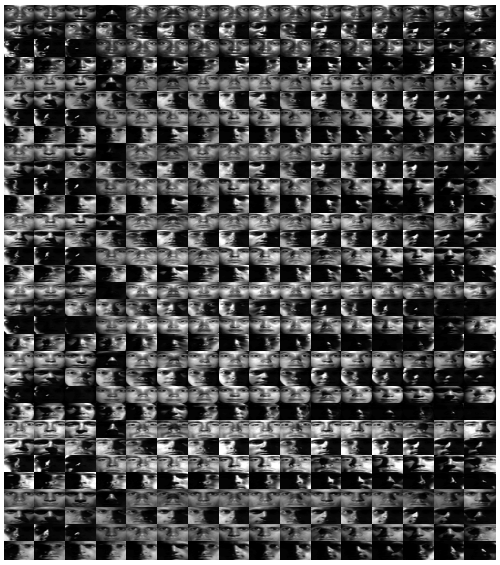
Algorithm 2 SSC with TV regularization

Input: A set of points $\{M_i\}_{i=1}^N$ (with missing entries) lying in a union of L linear subspaces $\{\mathcal{S}_\ell\}_{\ell=1}^L$
Output: Clustering and recovering results of the images
1: Normalization: normalize the data points.
2: SSC-TV: solve the SSC-TV optimization problem (4) by Algorithm 1.
3: Post-processing: for each C_i, keep its largest T coefficients in absolute magnitude, and set the remaining coefficients to zeros.
4: Similarity graph: form a similarity graph with N nodes representing the data points, and set the weights on the edges between the nodes by $W = |C| + |C|^T$.
5: Clustering: apply the normalized spectral clustering approach in [17] to the similarity graph.

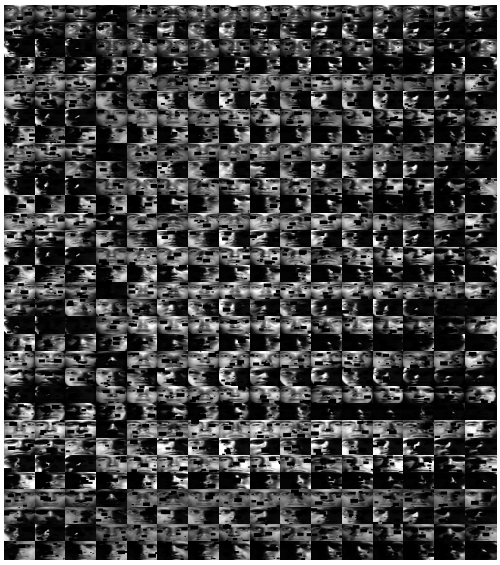
5.2. Implementation Details

We applied our Algorithm 2 to three public datasets: the Extended Yale B dataset[1], the USPS digital images dataset[2], and the COIL20 dataset[3]. The Extended Yale B dataset is a well-known dataset for face clustering, which consists of images taken from 38 human subjects, and 64 frontal images for each subject were acquired under different illumination conditions and a fixed pose. To reduce the computational cost and the memory requirements of the algorithms, we downsampled the raw images into the size of 48×42. Thus, each image is in dimension of 2,016. The USPS dataset is relatively difficult to handle, in which there are 7,291 labeled observations and each observation is a digit of 16×16 grayscale image and of different orientations.

[1] http://vision.ucsd.edu/~leekc/ExtYaleDatabase/ExtYaleB.html
[2] http://statweb.stanford.edu/~tibs/ElemStatLearn/data.html
[3] http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php



(a) Original images



(b) Incomplete images

(c) Recovered and clustered images

(d) Misclassified images

Figure 1: Sparse subspace clustering for incomplete face images from Extended Yale B dataset

Table 1: Results of different algorithms with different fraction of missing entries for different datasets

	spa=10%			spa=20%			spa=30%		
Alg.	SSC-0	SSC-AVE	SSC-TV	SSC-0	SSC-AVE	SSC-TV	SSC-0	SSC-AVE	SSC-TV
Yale B dataset: 3 subjects									
SCE-Mean	5.83	5.20	2.41	27.00	7.57	4.01	51.84	14.68	5.02
SCE-Median	0.85	0.81	0.78	31.51	5.20	1.04	57.29	11.71	2.60
Yale B dataset: 5 subjects									
SCE-Mean	5.62	4.76	4.31	40.32	17.23	5.29	54.28	32.23	7.75
SCE-Median	2.74	3.43	2.34	44.53	16.71	3.75	50.93	31.25	3.28
Yale B dataset: 8 subjects									
SCE-Mean	7.81	6.45	3.12	42.18	40.37	4.42	60.62	46.57	9.19
SCE-Median	4.75	5.89	3.32	40.13	41.30	3.22	60.44	45.80	4.88
USPS dataset: 3 subjects									
SCE-Mean	0.07	0.07	0.03	9.07	9.33	5.43	10.52	9.97	6.32
SCE-Median	0.00	0.00	0.00	1.67	0.67	0.67	1.67	2.17	1.50
USPS dataset: 5 subjects									
SCE-Mean	8.76	7.68	0.07	23.51	21.69	15.91	24.55	21.06	18.93
SCE-Median	0.40	0.40	0.00	28.00	26.60	16.20	25.70	24.70	26.20
USPS dataset: 8 subjects									
SCE-Mean	12.50	10.11	0.95	36.81	34.59	19.26	46.93	29.08	24.66
SCE-Median	17.38	15.84	0.13	37.31	35.63	19.06	50.13	27.63	21.13
COIL20 dataset: 3 subjects									
SCE-Mean	6.99	6.74	3.45	38.47	19.75	4.17	45.74	19.72	5.97
SCE-Median	0.00	0.00	0.00	50.93	2.55	0.00	51.39	5.56	0.00
COIL20 dataset: 5 subjects									
SCE-Mean	7.79	9.17	4.14	41.76	18.60	4.50	46.74	26.36	4.18
SCE-Median	0.00	0.00	0.00	37.22	5.69	0.00	46.11	28.33	0.00
COIL20 dataset: 8 subjects									
SCE-Mean	11.55	9.44	7.40	41.94	28.32	7.53	48.51	35.54	7.40
SCE-Median	10.33	0.00	0.19	40.80	20.89	3.04	46.96	45.83	2.69

The number of each digit varies from 542 to 1,194. To reduce the time and memory cost of the expriement, we randomly chose 100 images of each digit in our experiment. COIL20 is a database consisting of 1,440 grayscale images of 20 objects. Images of the 20 objects were taken at pose intervals of 5 degrees, which results in 72 images per object. All 1,440 normalized images of 20 objects are used in our experiment.

To study the effect of the number of subspaces in the clustering and recovery performance of algorithms, we applied algorithms under cases of $L = 3, 5, 8$, where L denotes the number of subspaces, i.e., the number of different subjects. To shorten the testing time, L subjects were chosen in the following way. For example, in the Extended Yale B dataset, all the 38 subjects were divided into four groups, where the four groups correspond to subjects 1 to 10, 11 to 20, 21 to 30, and 31 to 38, respectively. L subjects were chosen from the same group. For example, when $L = 5$, the number of possible 5 subjects is $3\binom{10}{5} + \binom{8}{5} = 812$. Among these 812 possible choices, 20 trials were randomly chosen to test the proposed algorithms under the condition of L subspaces.

These datasets were randomly corrupted to challenge the clustering and recovery ability of the proposed algorithms. To generate corrupted images with a specified missing fraction which is denoted as "spa", we randomly removed squares whose size is not larger than 10×10 in uniform manner, repeatedly until the total fraction of missing pixels is no less than the specified "spa" . To study the effect of the fraction of missing entries in clustering and recovery performance of algorithms, we artificially corrupted images into 3 different missing levels: 10%, 20% and 30%. To make the comparison of different algorithms as fair as possible, we randomly generated the missing images first, and then all algorithms were applied to the same randomly corrupted images to cluster the images and recover the missing pixels.

We use the following measure to demonstrate the clustering quality: the subspace clustering error (SCE)

$$SCE := (\#\text{ of misclassified images})/(\text{total }\#\text{ of images}).$$

For each set of L with different percentage of missing pixels, the averaged SCE over 20 random trials are calculated.

In Algorithm 1 (SSC-TV), we initiate Y by filling in each missing pixel with the average value of corresponding pixels in other images with known pixel value. We implement SSC proposed in [8] in two different ways. In Algorithm "SSC-AVE", we fill in the missing entries in the same way as "SSC-TV", and in Algorithm "SSC-0", we fill in the missing entries by 0.

In Algorithm 1, we choose $\beta = 5 \times 10^4$, $\gamma = 5 \times 10^4$, $\tau_1 = \tau_2 = 10^{-6}$ in all experiments. We set $\rho = 1000/\mu$, where $\mu := \min_i \max_{j \neq i} |Y_i^T Y_j|$ to avoid trivial solutions. It should be noted that ρ actually changes in each iteration, because matrix Y is updated in each iteration. We choose the parameter T in the post-processing procedure in Algorithm 2 as $T = 5$. To accelerate the convergence of Algorithm 1, we adopted the following stopping criterion to terminate the algorithm.

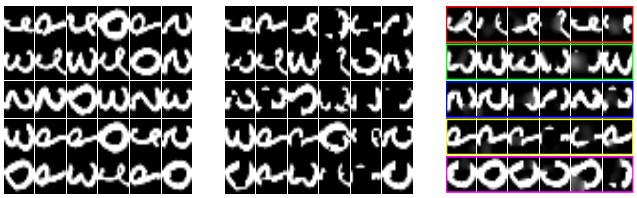
Note that we need to update both Y and C in Algorithm 1. To speed up the convergence, we stopped updating

(a) Original images (b) Incomplete images (c) Recovered images

(d) Original images (e) Incomplete images (f) Recovered images

(g) Original images (h) Incomplete images (i) Recovered images

Figure 2: Sparse subspace clustering for incomplete face images from Extended Yale B dataset. The 1st, 2nd and 3rd rows show the recovering and clustering results for incomplete images with 10%, 20%, and 30% pixels missing, respectively.



(a) Original images (b) Incomplete images (c) Recovered images

(d) Original images (e) Incomplete images (f) Recovered images

(g) Original images (h) Incomplete images (i) Recovered images

Figure 3: Sparse subspace clustering for incomplete face images from USPS dataset. The 1st, 2nd and 3rd rows show the recovering and clustering results for incomplete images with 10%, 20%, and 30% pixels missing, respectively.

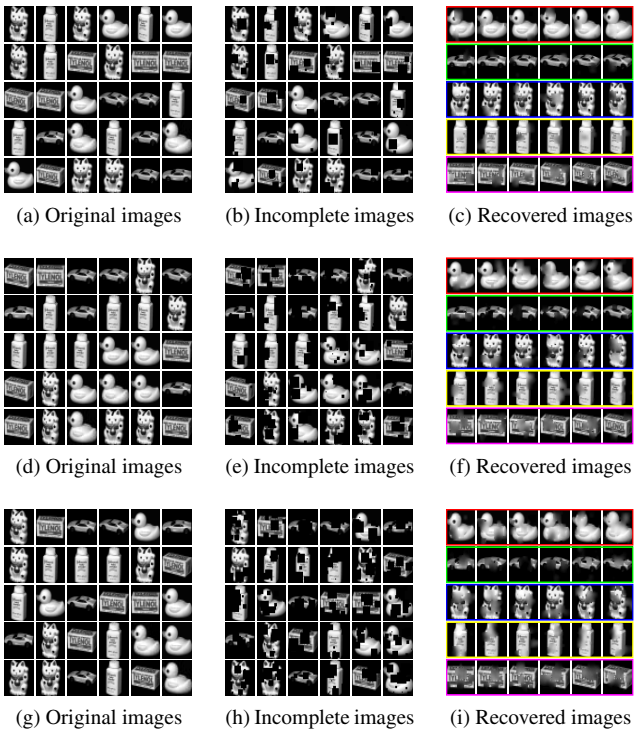
Y when $\frac{\mathrm{TV}_0 - \mathrm{TV}_k}{\mathrm{TV}_0 - \alpha * L * 64} \geq 0.8$, where TV_0 is the TV of the normalized incomplete images with missing entries, TV_k is the TV after k iterations. α is related to the mean of the TV terms of the images, and is chosen empirically. In our experiments, we chose $\alpha = 7.5$ for the extended Yale B dataset, $\alpha = 7.7$ for USPS dataset and $\alpha = 6.8$ for COIL20. Thus, when the above stopping criterion is satisfied, it indicates that the images have been restored well, and we can stop updating Y. Algorithm 1 is terminated when $\|(Y^{k+1}; C^{k+1}) - (Y^k; C^k)\|_F / \|(Y^k; C^k)\|_F \leq 2 \times 10^{-4}$.

5.3. Results

We report the experimental results in Table 1. From Table 1, we can see that when the images are incomplete, for example when $L = 3$, the mean of SCE is usually smaller than 7%. This means that the percentage of misclassified images is smaller than 7%. It can been seen that our Algorithm 2 gives better results on clustering errors in almost all situations. Especially, when $spa = 20\%$ and 30%, the mean clustering errors are much smaller than the ones given by SSC-AVE and SSC-0, this phenomena is more obvious with the increase of number of subjects. These comparison results show that our SSC-TV model can cluster the incomplete images very robustly and greatly outperforms SSC.

Figure 1 shows the clustering and recovery results of one instance of $L = 8$ using Algorithm 2 for the Extended Yale B dataset. The subfigure (a) are the original images before clustering. There are 512 images with 8 individuals, and 64 frontal images for each individual were acquired under different illumination conditions and a fixed pose. The subfigure (b) are incomplete images with 10% pixels missing. The subfigure (c) are the clustered and recovered images by our Algorithm 2. The misclassified images in each cluster are labeled with colored rectangles and the true positive rate (TP) and false positive rate are also given. The subfigure (d) are the misclassified images, and we can see that most of these images are not in good illumination conditions and they are thus difficult to be classified. Due to the space limit, we show the results of one instance of $L = 8$ for the USPS dataset and the COIL20 dataset in the supplementary material.

Figure 2 shows the clustering and recovery results of one instance of $L = 3$ using Algorithm 2 for the Extended Yale



(a) Original images (b) Incomplete images (c) Recovered images

(d) Original images (e) Incomplete images (f) Recovered images

(g) Original images (h) Incomplete images (i) Recovered images

Figure 4: Sparse subspace clustering for incomplete face images from COIL dataset. The 1st, 2nd and 3rd rows show the recovering and clustering results for incomplete images with 10%, 20%, and 30% pixels missing, respectively.

B dataset. To make the results more visually, we selected 6 images from each subject to show the clustering and recovery results. Part (a) of Figure 2 shows the 18 images taken from 3 subjects. They are ordered randomly because we do not know their clusters. Part (b) of Figure 2 shows the same images with 10% pixels randomly missing. Part (c) of Figure 2 gives the clustered and recovered images provided by the results of Algorithm 2. From Figure 2 we see that the 18 images are clustered into the correct clusters, and at the same time, the missing pixels are filled in and the images are recovered very well. The images in the second and third rows of Figure 2 respectively show the recovering and clustering results for incomplete images with 20% and 30% pixels missing. In Figures 3 and 4 we show one instance of $L = 5$ for the USPS dataset and the COIL20 dataset. They both show that our SSC-TV method can cluster and recover the incomplete images very well.

5.4. Choices of Post-Processing Parameter T

In Algorithm 2, a very important parameter is the thresholding parameter T in the post-processing procedure. This post-processing procedure was suggested in many existing

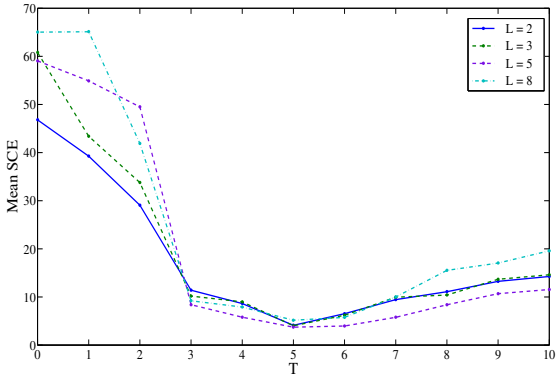


Figure 5: Effects of different T on Extended Yale B dataset

works (see, e.g., [6, 7]). However, the choice of T was not thoroughly investigated. In this section, we show the effects of different choices of T using Algorithm 2 on Extended Yale B dataset. We set the percentage of missing pixels as 10%, and vary T from 0 to 10, where $T = 0$ indicates that no post-processing procedure is performed. We plot the mean of SCE on the same dataset as in the previous section for different T in Figure 5. From Figure 5 we can see, for different number of subjects L, the averaged SCE by Algorithm 2 approximately achieves the minimum value at $T = 5$. That is to say, keeping the largest five components for each coefficient vector C_i is a good choice. This implies that each image in the subspace should be approximately a linear combination of five images in the same subspace. This observation is consistent with the conclusion in [19], where it is suggested that the face images taken from the same subject under different lighting conditions can be well-approximated by a five-dimensional subspace. Due to the space limit, we only show the choices of Post-Processing Parameter T on the Extended Yale B dataset. The similar results are also obtained on the other datasets.

6. Conclusion

In this paper, we proposed a novel subspace clustering approach for clustering incomplete images. Our approach integrates the techniques of sparse subspace clustering and total variation regularization, which enables to simultaneously recover and cluster incomplete images. The main ingredient of our approach is a nonconvex optimization model that obtains the affinity matrix of input data for spectral clustering. We developed an efficient first-order algorithm to solve the nonconvex optimization, and presented the convergence analysis under standard assumptions. The experimental results on images from the Extended Yale B dataset, the USPS dataset, and the COIL20 dataset have demonstrated that our approach works very well in comparison to the existing clustering methods.

References

[1] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. 4

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3:1–122, 2011. 3

[3] P. Bradley and O. L. Mangasarian. k-plane clustering. *Journal of Global Optimization*, 16:23–32, 2000. 1

[4] G. Chen and G. Lerman. Spectral curvature clustering (scc). *Int. J. Comput. Vision*, 81(3):317–330, Mar. 2009. 1

[5] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications in Pure and Applied Mathematics*, 57:1413–1457, 2004. 4

[6] E. L. Dyer, A. C. Sankaranarayanan, and R. G. Baraniuk. Greedy feature selection for subspace clustering. *arXiv preprint arXiv:1303.4778*, 2013. 4, 8

[7] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2790–2797. IEEE, 2009. 2, 8

[8] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2013. 1, 2, 4, 6

[9] B. Eriksson, L. Balzano, and R. Nowak. High rank matrix completion and subspace clustering with missing data. In *AISTATS*, 2012. 2

[10] J. Ho, M.-H. Yang, J. Lim, K.-C. Lee, and D. Kriegman. Clustering appearances of objects under varying illumination conditions. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–11. IEEE, 2003. 1

[11] J. J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994. 2

[12] H.-P. Kriegel, P. Kroger, and A. Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM Trans. Knowl. Discov. Data*, 3(1):1:1–1:58, 2009. 1

[13] 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. 2

[14] G. Liu, H. Xu, and S. Yan. Exact subspace segmentation and outlier detection by low-rank representation. In *Journal of Machine Learning Research*, 2011. 1

[15] Y. Lu and M. Do. Sampling signals from a union of subspaces. *Signal Processing Magazine, IEEE*, 25(2):41–47, 2008. 1

[16] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library (coil-20). Technical Report CUCS-005-96, Department of Computer Science, Columbia University, February 1996. 2

[17] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, pages 849–856, 2001. 2, 4

[18] A. Petukhov and I. Kozlov. Fast greedy algorithm for subspace clustering from corrupted and incomplete data. *CoRR*, abs/1306.1716, 2013. 2

[19] R. Ramamoorthi. Analytic pca construction for theoretical analysis of lighting variability in images of a lambertian object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(10):1322–1333, 2002. 1, 8

[20] S. R. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories. In *CVPR*, pages 1–8, 2008. 1

[21] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992. 2, 3

[22] Y. Shen, Z. Wen, and Y. Zhang. Augmented lagrangian alternating direction method for matrix separation based on low-rank factorization. *Optimization Methods and Software*, pages 1–25, 2012. 4

[23] M. Soltanolkotabi and E. J. Candes. A geometric analysis of subspace clustering with outliers. *The Annals of Statistics*, 40(4):2195–2238, 2012. 1

[24] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (gpca). In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'03, pages 621–628, Washington, DC, USA, 2003. IEEE Computer Society. 1

[25] Y.-X. Wang and H. Xu. Noisy sparse subspace clustering. In *ICML*, pages 89–97, 2013. 2

[26] Y. Xu, W. Yin, Z. Wen, and Y. Zhang. An alternating direction algorithm for matrix completion with nonnegative factors. *Frontiers of Mathematics in China*, 7(2):365–384, 2012. 4

[27] J. Yang, W. Yin, Y. Zhang, and Y. Wang. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM Journal on Imaging Sciences*, 2(2):569–592, 2009. 4

[28] J. Yang and Y. Zhang. Alternating direction algorithms for ℓ_1-problems in compressive sensing. *SIAM Journal on Scientific Computing*, 33(1):250–278, 2011. 4