Computer Vision and Image Understanding 138 (2015) 25–37

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

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

Reweighted sparse subspace clustering ☆

Jun Xu [a,c], Kui Xu [b], Ke Chen [b,*], Jishou Ruan [c]

[a] Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
[b] School of Computer Science and Software Engineering, Tianjin Polytechnic University, Tianjin, China
[c] School of Mathematical Sciences, Nankai University, Tianjin, China

ARTICLE INFO

Article history:
Received 30 September 2014
Accepted 15 April 2015
Available online 21 April 2015

Keywords:
Subspace clustering
Sparse representation
ℓ_1 minimization
Compressed sensing
Iterative weighting
Convex programming
Spectral clustering
Motion segmentation
Non-rigid motions
Human face clustering

ABSTRACT

Motion segmentation and human face clustering are two fundamental problems in computer vision. The state-of-the-art algorithms employ the subspace clustering scheme when processing the two problems. Among these algorithms, sparse subspace clustering (SSC) achieves the state-of-the-art clustering performance via solving a ℓ_1 minimization problem and employing the spectral clustering technique for clustering data points into different subspaces. In this paper, we propose an iterative weighting (reweighted) ℓ_1 minimization framework which largely improves the performance of the traditional ℓ_1 minimization framework. The reweighted ℓ_1 minimization framework makes a better approximation to the ℓ_0 minimization than tradition ℓ_1 minimization framework. Following the reweighted ℓ_1 minimization framework, we propose a new subspace clustering algorithm, namely, reweighted sparse subspace clustering (RSSC). Through an extensive evaluation on three benchmark datasets, we demonstrate that the proposed RSSC algorithm significantly reduces the clustering errors over the SSC algorithm while the additional reweighted step has a moderate impact on the computational cost. The proposed RSSC also achieves lowest clustering errors among recently proposed algorithms. On the other hand, as majority of the algorithms were evaluated on the Hopkins155 dataset, which is insufficient of non-rigid motion sequences, the dataset can hardly reflect the ability of the existing algorithms on processing non-rigid motion segmentation. Therefore, we evaluate the performance of the proposed RSSC and state-of-the-art algorithms on the Freiburg-Berkeley Motion Segmentation Dataset, which mainly contains non-rigid motion sequences. The performance of these state-of-the-art algorithms, as well as RSSC, will drop dramatically on this dataset with mostly non-rigid motion sequences. Though the proposed RSSC achieves the better performance than other algorithms, the results suggest that novel algorithms that focus on segmentation of non-rigid motions are still in need.

1. Introduction

In many real applications, high-dimensional data in several classes or categories can be respectively represented by corresponding low-dimensional subspaces. For example, motion trajectories of multiple rigidly moving objects in a video [1], face images of different subjects under varying illumination [2] all lie in low-dimensional subspaces of the ambient high-dimensional space. Subspace clustering refers to the task of separating the high-dimensional data into multiple low-dimensional subspaces according to their latent common patterns being recognized. Specifically, for a collection of $\{y_i\}_{i=1}^n$ points in $\mathbb{R}^m$, lying in a union of L subspaces, $\{S_j\}_{j=1}^L$ of dimensions $\{d_j\}_{j=1}^L$, while which points belong to which subspaces are unknown. The goal of subspace clustering is to identify the clustering of data so that points in the same cluster belong to the same subspace and find the parameters of each subspace. Such a model is an extension of the single subspace model found in many papers [3,4]. A more detailed definition of subspace clustering problem can be found in [5].

Subspace clustering has numerous applications in computer vision and image processing, e.g., motion segmentation [6,7] and face clustering [8,9]. The motion segmentation problem refers to segmenting the motion trajectories of different objects from tracked points in video sequences which are captured by a static or moving camera [10,11]. The face clustering problem refers to clustering the face images of multiple subjects according to their face images acquired with a fixed pose but varying illumination. Recently, the subspace clustering problem has drawn attention of researchers in compressed sensing, which is a hot research area in information science [12,13].

☆ This paper has been recommended for acceptance by David Jacobs.

* Corresponding author.
E-mail addresses: csjunxu@comp.polyu.edu.hk (J. Xu), kuixu.tj@gmail.com (K. Xu), kchen1.tj@gmail.com (K. Chen), jsruan@nankai.edu.cn (J. Ruan).

http://dx.doi.org/10.1016/j.cviu.2015.04.003
1077-3142/

1.1. Prior work on subspace clustering

Numerous subspace clustering algorithms have been proposed in the past [6,9,10,14–29]. According to the mathematical framework they employ, existing subspace clustering algorithms can be divided into three main categories: algebraic, statistical, and spectral clustering [9].

The algebraic algorithms solve the subspace clustering problem by modeling a subspace as a gradient of a polynomial [11,14]. These methods do not require prior information of each subspace, and can enforce structural restriction on the subspaces. Shape interaction matrix (SIM) [6] and generalized principal component analysis (GPCA) [14] are two classical methods which belong to this category. Though having a lot of advantages, the algebraic algorithms are hard to avoid exponentially expensive computations due to the polynomial fitting. Furthermore, these algorithms are generally sensitive to noise and outliers and unable to resolve the difficulty of clustering points near the intersection of subspaces, and have exponentially complex computation with respect number and dimensions of subspaces [9,22].

The statistical algorithms probabilistically model each subspace as a Gaussian distribution, and consider the clustering problem as an estimation of mixture of Gaussian. Specific algorithms include agglomerative lossy compression (ALC) [10], mixture of probabilistic PCA (MPPCA) [15], multi-stage learning (MSL) [16], and the robust method known as RANSAC [17]. These algorithms typically require prior information of the subspaces, such as the number of subspaces and their dimensions. The computational complexity of the above mentioned algorithms is also exponential with respect to the number of subspaces and their dimensions [9,22].

The spectral clustering algorithms use local information around each data point to build a similarity between pairs of points. The clustering of data points is achieved by applying spectral clustering to the affinity matrix. Local subspace affinity (LSA) [18], spectral local best-fit flats (SLBF) [19], locally linear manifold clustering (LLMC) [20], and spectral curvature clustering (SCC) [21] are methods of this class. They cannot deal well with points near the intersection of two subspaces if the neighborhood of this point contains points from different subspaces. Inspired by the emerging field of compressed sensing (CS) [12,13], the sparse subspace clustering (SSC) algorithm [9,22] solves the clustering problem by seeking a sparse representation of data points used as a dictionary. Hence, by resolving all the sparse representations for all data points and constructing an affinity graph, SSC automatically finds different subspaces as well as their dimensions from a union of subspaces. Finally, the subspace clustering is performed by spectral clustering [30]. In addition, as ℓ_1-norm minimization is convex and needs at most polynomial time in complexity, SSC deals well with the data. A robust version of SSC to deal with noise and corruptions or missing observations is also given in [9,22]. Instead of finding a sparse representation, the low-rank representation (LRR) algorithm [23,24] poses the subspace clustering problem as finding a low-rank representation of the data over the data itself. Then, Lu et al. proposed a method based on least squares regression(LSR) [28] which takes advantage of data correlation and groups highly correlated data together. The grouping information can be used to construct an affinity matrix which is block diagonal and can be used for subspace clustering through spectral clustering algorithms. Recently, Lin et al. analyzed the grouping effect deeply and proposed the smooth representation framework (SMR) [29] which also achieves state-of-the-art performance in subspace clustering problem. Different from SSC, the LRR, LSR and SMR algorithms use normalized cuts [31] in the spectral clustering step.

Currently, SSC, LRR, LSR and SMR achieve state-of-the-art performance on subspace clustering problem than the other methods [6,10,14–21,25–27]. This can be demonstrated by the clustering performance of these methods on benchmark datasets such as the Hopkins155 dataset [7] and Extended Yale B dataset [8]. However, the performance of these four state-of-the-art algorithms [9,22–24,28,29] on motion segmentation of non-rigid moving objects is not thoroughly revealed as the Hopkins155 dataset [7] has its bias [32]. This dataset only includes a few non-rigid motion sequences. On the other hand, the performance of these algorithms on face clustering problem still has a large space to improve.

1.2. Paper contribution

In our work, we will first theoretically demonstrate the advancement of reweighted ℓ_1-norm minimization framework over ℓ_1 minimization. From recent work [33–37] in the compressed sensing field, we speculate that the performance of SSC [9,22] can be largely improved if we use iterative weighting (i.e., reweighted) ℓ_1 minimization framework instead of ℓ_1-norm minimization. The improvements can be applied into many subspace clustering problems (motion segmentation, face clustering, and face recognition [9,22–24,38]). Our main contributions are twofold. First, through a series of experiments on the Hopkins155 dataset [7] and Extended Yale B dataset [8], we demonstrate that our method largely reduces the clustering errors with little additional computational cost. Second, we found the limitation of the representation-based methods in subspace clustering, especially in non-rigid motion segmentation. We test the state-of-the-art algorithms [9,18,21,24,28,29]] and our proposed RSSC on a different motion segmentation dataset: the Freiburg-Berkeley Motion Segmentation Dataset [32,39] (for more details, please see the experiments section). We divide this dataset into two parts: rigid motion and non-rigid motion. All these algorithms will get good performance on the part of rigid motions while achieve high clustering errors on the part of non-rigid motions, though our method will achieve the lowest clustering error on this dataset.

The paper is organized as follows: Section 2 briefly introduces the ℓ_1 minimization framework and reweighted ℓ_1 minimization framework. Section 3 introduces the SSC algorithm and the proposed RSSC algorithm. Section 4 presents our experimental results on the Hopkins155 dataset [7], the Freiburg-Berkeley Motion Segmentation Dataset [32,39], and Extended Yale B database [8]. Section 5 concludes this paper and discuss some future work.

2. Reweighted ℓ_1 minimization

2.1. The ℓ_1 minimization framework

The use of ℓ_1 minimization can be traced back to the year 1973 [40]. It is first applied in reflection seismology [40–42]. After that, the nature of sparsity of the ℓ_1 minimization was confirmed and it began to be used in signal recovery [43,44] and image processing [45]. The famous LASSO algorithm [46] and Basis Pursuit [47] are just two of numerous applications of the ℓ_1 minimization. Due to its wide applicability, the ℓ_1 regularization is considered the “modern least squares”. For more details of the progress of the ℓ_1 minimization, please refer to [37].

In mathematics, many problems in signal recovery and image processing need to identify the sparsest solution of an underdetermined linear system. For example, given an $m \times n$ matrix $\boldsymbol{A}$ with $m \le n$ and a nonzero vector $\boldsymbol{b} \in \mathbb{R}^m$, to find a sparsest solution of $\boldsymbol{Ax} = \boldsymbol{b}$ equals to solving the ℓ_0 minimization problem

$$(P_0) \quad \min_{\boldsymbol{x} \in \mathbb{R}^n} \|\boldsymbol{x}\|_0 \quad \text{s.t.} \quad \boldsymbol{A}x = \boldsymbol{b}, \tag{1}$$

where $\|\boldsymbol{x}\|_0$ is the number of nonzero components of $\boldsymbol{x}$. Since the problem (P_0) is a nonconvex and NP hard optimization problem [48], approximate solutions are considered instead. In the past decade, several greedy pursuit algorithms have been proposed, such as the Matching Pursuit (MP) [49] and the Orthogonal Matching Pursuit (OMP) algorithms [50]. Another pursuit algorithm is the Basis Pursuit