

Multiple Subspaces Separation in Case of Camera Motion

Salehe Erfanian Ebadi, Ebroul Izquierdo

Queen Mary University of London, United Kingdom

{s.erfanianebadi, e.izquierdo}@qmul.ac.uk

Keywords: Subspace clustering, approximated RPCA, camera motion, trajectory decomposition,

Abstract

We explore the problem of subspace clustering. Given a set of data samples approximately drawn from a union of multiple subspaces, our goal is to cluster the samples into respective subspaces, and also remove possible outliers. We propose an Approximated Robust PCA Clustering (ARPCAC) method that involves extracting the point trajectories only induced by object motion, from the pool of all motions induced by objects and camera motion, and then projecting them onto a 5-dimensional space, using PowerFactorization. Our algorithm can be used to segment multiple motions in video and furthermore, is extended to the problem of face clustering. Conducted experiments demonstrate state-of-the-art performance.

1 Introduction

Several types of visual data, such as motion, face, and texture, have been known to be well characterized by subspaces. Recently, there has been an increasing interest on the geometrical and statistical models for the understanding of *dynamic* scenes, in which both the camera and multiple objects move. The widely used Principal Component Analysis (PCA) method and the recently established matrix completion and recovery methods are essentially based on the hypothesis that the data is approximately drawn from a low-rank subspace. However a given dataset can seldom be well described by a *single* subspace. A more reasonable model is to consider data as lying near *several* subspaces. When the data is clean, i.e., the samples are strictly drawn from the subspaces, several existing methods (e.g., [3, 5, 17]) are able to exactly solve the subspace segmentation problem. So, the main challenge of subspace segmentation is to handle the *errors* (e.g., noise and corruptions) that possibly exist in data, i.e., to handle the data that may not strictly follow subspace structures. With this outlook, we study the following *robust subspace clustering* problem: Given a set of data samples approximately drawn from a union of linear subspaces, correct the possible errors and segment all samples into their respective subspaces, and simultaneously reveal each subspace's independent motion. By independent motion we mean the camera-induced motion has been subtracted from all the motions in the scene, where the motion trajectory of an object can be revealed. Two main applications, motion seg-

mentation and face clustering are studied in this paper for this problem.

We propose a novel method termed ARPCAC (Approximated Robust Principal Component Analysis Clustering). Given a set of data samples, each of which can be represented as a combination of a low-rank and sparse subspace, ARPCAC aims at finding the *lowest rank* representation of all data jointly, while simultaneously revealing the independent motion of each subspace. The computational procedure of ARPCAC is to solve a *Frobenius* and $\ell_{2,1}$ -norm regularized optimization problem, which is convex and can be solved in polynomial time. It can be shown that the ARPCAC can well solve the subspace clustering problem. The subspace membership is provably determined by belonging to either of the low-rank, sparse, or error patterns, and hence the ARPCAC can perform robust subspace clustering and error correction in an efficient way. Motion segmentation from multiple views has been studied in the case of affine cameras, because in this case the motion of each one of the rigidly moving objects lives in a four-dimensional subspace [5]. In this paper however, we do not need to assume an affine camera model, since the camera motion will be compensated for by the dominant subspace that is reasonably close to the background motion in most practical applications.

2 Related Work

Mixture of Gaussian has been used in [12] where a maximum likelihood estimate was used, and in [10] where Random Sample Consensus (RANSAC) were adopted. These methods are sensitive to errors, and this problem is still not well solved due to optimization difficulty. Factorization-based methods [3] seek to approximate the given data matrix as a product of two matrices such that the support pattern for one of the factors reveals the segmentation of the samples. Generalized Principal Component Analysis (GPCA) [18] presents an algebraic way to model the data drawn from a union of multiple subspaces. However, this method is sensitive to noise due to difficulty of estimating the polynomials from real data. Subspace segmentation has also been regarded as a clustering problem, where an affinity matrix is learned to obtain the final segmentation results by spectral clustering (SC) algorithms such as Sparse Subspace Clustering (SSC) [5], the LRR [17], and the proposed ARPCAC method. The main difference is the approach for learning the affinity matrix.

3 Low-Rank Modeling of Samples

In this section, we present the ARPCAC method for recovering a matrix from corrupted and incomplete observations. Let D be a collection of data samples in presence of outliers and corruptions. That is, for the set of points $X_p \in P^3$ in frame $f \in F$, we can stack all the image measurements into a $2F \times P$ matrix D as

$$D = \begin{bmatrix} X_1^1 & \dots & X_P^1 \\ Y_1^1 & \dots & Y_P^1 \\ \vdots & \ddots & \vdots \\ X_1^F & \dots & X_P^F \\ Y_1^F & \dots & Y_P^F \end{bmatrix} \quad (1)$$

In order to recover the low-rank matrix L from the given observation matrix D corrupted by errors E it is straightforward to consider the following regularized rank minimization [6]

$$\min_{\text{rank}(L) \leq r, E, \tau} \|D \circ \tau - L - E\|_F^2 + \lambda \|E\|_{2,1} \quad (2)$$

$$\text{s.t. } D \circ \tau = L + E,$$

where $\text{rank}(L) \leq r \ll \text{rank}(D)$, and $\lambda > 0$ is a parameter, $\|\cdot\|_F$ is the Frobenius-norm, and $\|\cdot\|_{2,1}$ is the $\ell_{2,1}$ -norm which is the ℓ_1 -norm of the vector formed by taking the ℓ_2 -norm of a matrix, and promotes group sparsity. The error pattern E can be described as combination of a sparse pattern S containing the underlying subspaces, and a noise pattern G that contains the noise, outliers, and incomplete samples. τ stands for some transformation in the image domain (e.g., 2D affine transformation for correcting misalignment, or 2D projective transformation for handling some perspective change in the camera model). And henceforth, $L \circ \tau$ describes the lowest rank estimate for samples drawn solely from the camera motion, whereas S describes all underlying subspaces, and G contains errors. From the sample set $L \circ \tau + S$ we can obtain reliable trajectories for all subspaces. The assumption here is that each subspace has a spectral nature, i.e., each subspace will form a unique affinity matrix that can be used to reveal the true segmentation of data. Also, $L \circ \tau$ provides the underlying lowest rank representation for the data that helps reduce the problem to a simple clustering of independent motions in the scene, as samples S are only drawn from object-induced trajectories.

The problem (2) is a difficult, non-convex optimization problem. Fortunately, we can find a good initialization by pre-aligning all frames in the sequences to the middle frame, before the main loops of minimization. The pre-alignment is done by the robust multi-resolution method proposed in [19]. This practice is successful in most cases given that a drastic scene change does not occur in the sequence. As described in [26], we can then solve (2) by repeatedly linearizing about the current estimate of τ , and seeking a deformation step $\Delta\tau$. In other words, at each iteration, we update τ by a small increment $\Delta\tau$ and linearize $A \circ \tau$ as $D \circ \tau + J\Delta\tau$, where J denotes the Jacobian matrix $J = \frac{\partial D}{\partial \tau}$. Thus, τ can be updated via the following minimization problem

$$\tau^t \leftarrow \tau + \arg \min_{\Delta\tau} \|D \circ \tau - L^{t-1} - S^{t-1} + J\Delta\tau\|_F^2 \quad (3)$$

The minimization over $\Delta\tau$ in (3) is a weighted least-squares problem which has a closed-form solution. In practice, the update of τ for each frame can be done separately since the transformation is applied on each image individually. Thus the update of τ is efficient. Then we proceed by using an alternating minimization procedure to solve L and S one at a time until the solution reaches convergence and show that it is efficient; that means solving two reduced problems, each being minimized independently from one another

$$L^t = \arg \min_{\text{rank}(L) \leq r} \|D \circ \tau - L - S^{t-1}\|_F^2 \quad (4)$$

$$S^t = \arg \min_S \|D \circ \tau - L^t - S\|_F^2 + \lambda \|S\|_{2,1} \quad (5)$$

The residual error of the approximation of D by $L \circ \tau + S$ is stored in G . The entries of G can be very large in magnitude, but random and scattered, exhibiting the behavior of error deviation as described. The discerning difference between S and G is that G shows no structure in the sparsity domain, that of which is determined by the $\ell_{2,1}$ -norm minimizer.

3.1 Independent Subspace Motion Extraction

The obtained trajectories in S are induced from two motion components: rigid camera motion, and object motion. When the motion of interest includes global object motion, it can be further decomposed into two components: rigid object motion, and articulated motion. We employ the latest advances in sparse optimization to estimate each of these components, and extract the object trajectories which solely correspond to the motion of interest. [27] and [20] have assumed that the majority of the observed motion is induced by the camera motion; this assumption will not fit most realistic data, so we refrain from doing so to not cause any loss of generality. Therefore, the trajectories drawn from samples should generally span a subspace determined by the scene structure and the camera's intrinsic and extrinsic parameters. In order to find the basis for the subspace trajectory, we have obtained a $2F \times P$ (P samples) matrix S from ARPCAC using the position vectors of the trajectories in a sequence

$$S = \begin{bmatrix} \mathcal{X}_1^1 & \dots & \mathcal{X}_P^1 \\ \mathcal{Y}_1^1 & \dots & \mathcal{Y}_P^1 \\ \vdots & \ddots & \vdots \\ \mathcal{X}_1^F & \dots & \mathcal{X}_P^F \\ \mathcal{Y}_1^F & \dots & \mathcal{Y}_P^F \end{bmatrix} \quad (6)$$

Through the following rank minimization surrogate, we can decompose S into two components: a low-rank matrix \mathcal{L} , and the sparse error matrix \mathcal{E}

$$\arg \min_{\mathcal{L}, \mathcal{E}} \|\mathcal{L}\|_* + \xi \|\mathcal{E}\|_1 \quad \text{s.t. } S = \mathcal{L} + \mathcal{E}, \quad (7)$$

with $\|\cdot\|_*$ defining nuclear-norm which is the sum of singular values $\|\mathcal{L}\|_* = \sum_i(\sigma_i)$, and $\|\cdot\|_1$ the ℓ_1 norm. ξ trades off the rank solution versus the sparsity of the error, and is always set to $1.1/\sqrt{P}$ following the theoretical considerations in [1]. The equation (7) can be solved with convex optimization

methods such as the Augmented Lagrange Multiplier (ALM) algorithm [16]. The columns of the resulting low-rank matrix \mathcal{L} define the basis of the low-rank components in the trajectories. The subspace spanned by the major basis of \mathcal{L} correspond to the desired background subspace which includes both the background trajectories and the camera motion component of the foreground (objects in the scene) trajectories. On the other hand, any rigid body motions in the scene will also contribute to \mathcal{L} ; therefore, the subspace spanned by the rest of the basis of \mathcal{L} mostly correspond to rigid body motions. Since the camera motion subspace is approximately spanned by three basis [12, 5], the camera motion component can be estimated by $\mathcal{L}_c = us^*v^T$, where u and v are obtained by singular value decomposition $[u, s, v] = SVD(\mathcal{L})$, and s^* is the top three most significant singular values of s . Therefore, the rigid body motion component is expressed by $\mathcal{L} - \mathcal{L}_c$. Moreover, the columns of the matrix \mathcal{E} correspond to the deviation of each trajectory from the recovered low-rank subspace, which captures the articulated motions [27]. Therefore, the total object trajectories \mathcal{E}_t that include the articulated and the rigid body motion is given by

$$\mathcal{E}_t = \mathcal{E} + \mathcal{L} - \mathcal{L}_c \quad (8)$$

Figure 1 shows the motion decomposition for two sequences from the Hopkins155 dataset. As it can be seen, the trajectories that are obtained for the background and foreground, are both contaminated by the camera motion. Note the motion trajectory of the woman walking in the middle column is completely different from the actual motion trajectory that is revealed by ARPCAC in the right column. Clean motion trajectories are crucial for applications such as human motion analysis, and the trajectories in the middle column – which is usually what is obtained by trajectory extractors – would adversely affect the results. In the next example we show that ARPCAC can cluster multiple independent motion subspaces. Figure 2 illustrates the motion decomposition for three examples in the Hopkins155 dataset. From these examples it is clear that the proposed independent object motion extractor is successful in subtracting camera motion from each motion subspace and simultaneously clustering each motion trajectory into its corresponding subspace.

4 Segmentation of Multiple Rigid-Body Motions

We can use a combination of ARPCAC and PowerFactorization that leads to the following geometric solution to the multi-frame 3D motion segmentation problem [25]. First, we project the motion trajectories obtained from object-induced motion \mathcal{E} extracted by ARPCAC onto a five-dimensional subspace using the PowerFactorization. Then, we fit a collection of subspaces to the projected trajectories, by fitting a homogeneous polynomial representing all motion subspaces to the projected data. Next, we obtain a basis for each motion subspace from the derivatives of this polynomial. And finally, we apply spectral clustering to a similarity built from the subspace angles. We have tested our approach on a database of 155 motion

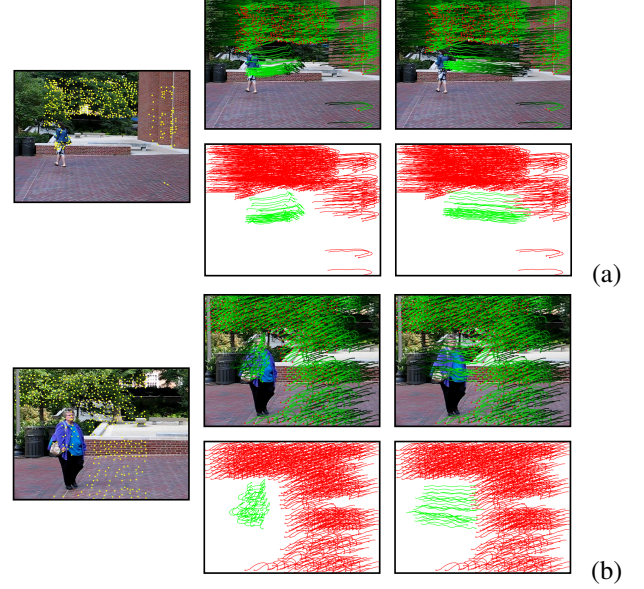


Figure 1. Two examples of independent subspace motion extraction in (a) and (b). Left: last frame in a sequence with trajectory particles. Middle: obtained trajectories from tracked motion samples (red corresponds to background subspace trajectories induced by camera motion \mathcal{L} , and green corresponds to foreground object trajectories induced by both camera motion and object motion \mathcal{S}). Right: extracted clean foreground object trajectories \mathcal{E} in green induced *only* by object motion, revealing the true object trajectory. In the second and third columns motion trajectories of the top figure are shown overlaid over white background in the bottom figure for better visualization. Please refer to supplementary video at <https://youtu.be/ndE1KZG3yrQ> for more examples.

sequences with full, independent, degenerate, dependent motions, missing data, outliers, etc. Our algorithm achieves error of 0.89% for two motions and 3.78% for three motions.

4.1 Projection using PowerFactorization

From here on, without loss of generality, we refer to the sample matrix as W that could refer to either object samples \mathcal{S} or object-induced samples \mathcal{E} . We wish to replace W by a matrix obtained by projecting its columns onto a 5-dimensional subspace. If AB^T is the nearest rank-5 factorization to W , then $\tilde{W} = B^T$ is the matrix that we require. The measure of closeness of AB^T to W is

$$\sum_{(i,j) \in \mathcal{I}} (W_{ij} - (AB^T)_{ij})^2, \quad (9)$$

where \mathcal{I} is the set of pairs (i, j) for which W_{ij} is known. With PowerFactorization we start with a random matrix A_0 , and alternate the following steps until convergence of $A_k B_k^T$. Essentially this algorithm alternates between computing A_k and B_k using least-squares.

1. Given A_{k-1} , find the $P \times r$ matrix B_k that minimizes $\sum_{(i,j) \in \mathcal{I}} |W_{ij} - (A_{k-1} B_k^T)_{ij}|^2$.
2. Orthonormalize the columns of B_k by replacing it by a matrix B'_k such that $B_k = B'_k N_k$, where B'_k has orthonormal columns, and N_k is upper-triangular.

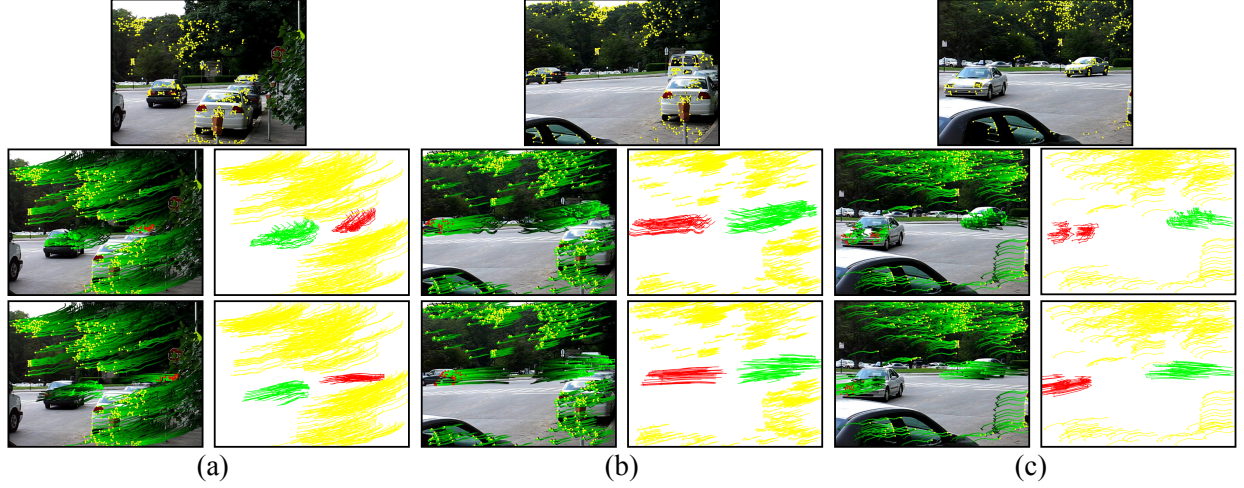


Figure 2. Three examples of independent subspace motion extraction (a), (b), and (c). Top: last frame in a sequence with trajectory particles. Middle: obtained trajectories from tracked motion samples (yellow corresponds to background subspace trajectories induced by camera motion L , red and green correspond to foreground object trajectories induced by both camera motion and object motion S). Bottom: extracted clean foreground object trajectories \mathcal{E} in red and green induced *only* by object motion, revealing the true object trajectory. For each example, motion trajectories of the left figure are shown overlaid over white background in the right figure for better visualization. Please refer to supplementary video at <https://youtu.be/ndE1KZG3yrQ> for more examples.

- Given B_k , find the matrix A_k that minimizes $\sum_{(i,j) \in \mathcal{I}} |W_{ij} - (A_k B_k^T)_{ij}|^2$

4.2 Fitting Polynomials to Projected Trajectories

We have reduced the motion segmentation problem to finding a set of linear subspaces in \mathbb{R}^5 , each of dimension at most 4, which contain the data points (or come close to them). The points in question, $\{w_p\}_{p=1}^P$, are the columns of the projected data matrix $\hat{W} = [w_1, \dots, w_P] \in \mathbb{R}^{5 \times P}$. We obtain a polynomial q representing the n motion subspaces by computing its vector of coefficients $c \in \mathbb{R}^{M_n}$ as the singular vector of the embedded data matrix $\tilde{W} = [\tilde{w}_1, \dots, \tilde{w}_P] \in \mathbb{R}^{M_n \times P}$ corresponding to its smallest singular value.

4.3 Feature Clustering via Polynomial Differentiation

The feature points can then be clustered by applying spectral clustering to the similarity matrix $S_{ij} = \cos^2(\theta_{ij})$, where θ_{ij} is the angle between the vectors $\nabla q(w_i)$ and $\nabla q(w_j)$ for $i, j = 1, \dots, P$, with the derivative of q defined as a 5-vector

$$\nabla q(w) = (\partial q / \partial w_1, \dots, \partial q / \partial w_5) \quad (10)$$

Then the standard factorization approach is applied to each one of the n group of features to obtain motion and structure parameters.

5 Experiments

In the experiments of this paper, we focus on analyzing the essential aspects of ARPCAC under the context of subspace segmentation and outlier detection. We have implemented our algorithm in MATLAB R2015a on a desktop machine with a Core i7-4770 (single core). For all the tests we chose $\lambda =$

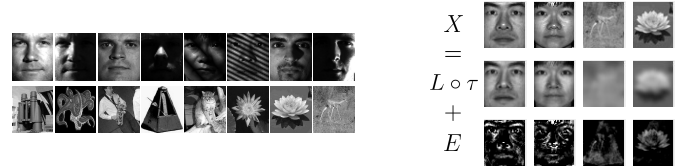


Figure 3. Left: examples of the images in the Yale-Caltech dataset same as the one used in [17]. Right: using ARPCAC to correct the errors in the Yale-Caltech dataset; from top to bottom: the original data matrix X , the corrected data $L \circ \tau$, the error E .

5×10^{-3} . We compare our method to some previous subspace segmentation methods including RANSAC [10], GPCA [18], LSA [29], ALC [21], SSC [5], SCC [2], MSL [23], LLMC [11], LBF [30], LRR [17], LRR-H [17], LRSC [7], RPCA methods from RPCA₁ [1], RPCA_{2,1} [28], and [22], SR [4], and SLBF [30], BDLRR [9], and the most recent work S³C [15].

5.1 Hopkins155

To verify the segmentation performance of ARPCAC, we adopt for experiments the Hopkins155 [24] motion database, which provides an extensive benchmark for testing various subspace segmentation algorithms. In Hopkins155, there are 155 video sequences along with the features extracted and tracked in all the frames. The segmentation performance for this dataset is shown in Table 1. These results illustrate that ARPCAC performs considerably better than other PCA-based counterparts, namely PCA, RPCA₁, RPCA_{2,1}, SR, LRR, and GPCA. Besides the superiority in segmentation accuracy, another advantage of ARPCAC is that it can work well under a wide range of parameter settings as we chose the same λ value for all the test, whereas other PCA-based methods except for LRR are



Figure 4. An example from the LFW database. Left: original images D ; middle: aligned images $D \circ \tau$; right: errors E .

Table 1. Segmentation Errors (%) on Hopkins155.

	PCA	RPCA ₁	RPCA _{2,1}	SR	LRR	GPCA	RANSAC	BDLRR	ARPCAC
mean %	4.56	4.13	3.26	3.89	1.59	10.34	9.76	4.33	1.53
	LLMC	LBF	ALC	SCC	SLBF	SSC	MSL	S ³ C	LSA
mean %	4.80	3.72	3.37	2.70	1.35	1.24	5.06	2.20	4.94

Table 2. Average run time (seconds) per sequence for segmentation task on Hopkins155 for RPCA-based methods.

PCA	RPCA ₁	RPCA _{2,1}	SR	LRR	ARPCAC
0.2	0.8	0.8	4.2	1.9	0.4

Table 3. Clustering Error (%) of Different Algorithms on Hopkins155 for 2 and 3 motions.

	LSA	SCC	LRR	LRR-H	LRSC	SSC	S ³ C	BDLRR	ARPCAC
2 Motions									
mean %	3.61	3.04	4.83	3.41	3.87	1.83	1.64	3.70	0.89
median %	0.51	0.00	0.26	0.00	0.26	0.00	0.00	0.00	0.00
3 Motions									
mean %	7.65	7.91	9.89	4.86	7.72	4.40	4.11	6.49	3.78
median %	1.27	1.14	6.22	1.47	3.80	0.00	0.73	1.20	1.31
All									
mean %	4.52	4.14	5.98	3.74	4.74	2.41	2.20	4.33	1.53
median %	0.57	0.00	0.59	0.00	0.58	0.00	0.00	0.00	0.00

sensitive to the parameter λ . As for comparison to state-of-the-art methods in the lower tier of Table 1, our method performs on par, and achieves third place after SSC and SLBF. This performance can be improved if λ is tuned per-problem, but we refrain from doing so as we would like to demonstrate an autonomous performance. Moreover, our algorithm is superior in clustering multiple motions in a scene as shown in Table 3, whereas, SSC and SLBF both are more well-suited for single motion segmentation. The efficiency in terms of running time of ARPCAC is comparable to PCA and surpasses that of other PCA-based methods as shown in Table 2, making it suitable for real-time performance. The results of applying subspace clustering algorithms to the dataset using the original $2F$ -dimensional feature trajectories for 2-motion and 3-motion categories on Hopkins155 are shown in Table 3. Our algorithm achieves top performance in all motion categories.

5.2 Yale-Caltech

To test ARPCAC’s effectiveness in the presence of outliers and corruptions, we create a dataset by combining the Extended Yale Database B [14] and Caltech101 [8]. Figure 3-Left shows

Table 4. Segmentation Accuracy (ACC) and time consumption comparison on Yale-Caltech for PCA-based methods.

	PCA	RPCA ₁	RPCA _{2,1}	SR	LRR	ARPCAC
Accuracy %	77.15	82.97	83.72	73.17	86.13	89.22
time (seconds)	0.6	60.8	59.2	383.5	152.6	53.87

some examples of this dataset. It can be seen in Table 4 that ARPCAC is better than PCA and RPCA methods in terms of both subspaces segmentation and outlier detection. To visualize ARPCAC’s effectiveness in error correction, Figure 3-Right shows some produced results. It is worth noting that the “error” term E can contain “useful” information, e.g., eyes and salient parts, that can be used for emotion and visual cue recognition. The low-rank part $L \circ \tau$ corresponds to the principal features of each subject that discriminate it from the rest of the data. The aligned and cleaned $L \circ \tau$ part can be used for face recognition, and face clustering as done in this paper.

5.3 LFW

The Labeled Faces in the Wild (LFW) database of public figures [13] exhibit significant variations in pose and facial expression, illumination, and occlusion; moreover, the ground truth (i.e., undistorted, not rotated, not shifted) image is not known. In total there are 681 samples of images taken from 20 subjects. Our ARPCAC aligns these images to a 80×60 canonical frame, and Affine transformations τ are used to cope with large variability in poses. Figure 4 shows one example from the results on this dataset. Our algorithm proves itself to be effective even in presence of large misalignment and corruptions.

6 Conclusion

We have proposed a low-rank and sparse representation to identify the subspace structures from corrupted data. Namely, our goal is to segment the samples into their respective subspaces and correct the possible errors simultaneously while revealing each subspace’s independent motion. ARPCAC is a generalization of the recently established RPCA method [1], extending the recovery of corrupted data from single subspace to multiple subspaces that are dynamic where both camera and the scene objects move. Both theoretical and experimental results show the effectiveness of ARPCAC in subspace segmentation and misaligned and corrupted face clustering applications. In future, we would like to extend ARPCAC to not rely on Power-

Factorization and Spectral Clustering for the separation of the extracted independent motions, by the low-rank and sparse decomposition.

Acknowledgements

The research leading to this paper was fully supported by the SafeShore project (<http://safeshore.eu/>) with funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 700643.

References

- [1] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *J. ACM*, 58(3):11:1–11:37, June 2011.
- [2] Guangliang Chen and Gilad Lerman. Spectral curvature clustering (scc). *International Journal of Computer Vision*, 81(3):317–330, 2009.
- [3] João Paulo Costeira and Takeo Kanade. A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29(3):159–179, 1998.
- [4] David L Donoho. For most large underdetermined systems of linear equations the minimal ℓ_1 -norm solution is also the sparsest solution. *Communications on pure and applied mathematics*, 59(6):797–829, 2006.
- [5] Ehsan Elhamifar and René Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(11):2765–2781, 2013.
- [6] Salehe Erfanian Ebadi and Ebroul Izquierdo. Foreground segmentation with tree-structured sparse RPCA. *Pattern Analysis and Machine Intelligence (PAMI), IEEE Transactions on*, 2017.
- [7] Paolo Favaro, René Vidal, and Avinash Ravichandran. A closed form solution to robust subspace estimation and clustering. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1801–1807. IEEE, 2011.
- [8] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.
- [9] Jiashi Feng, Zhouchen Lin, Huan Xu, and Shuicheng Yan. Robust subspace segmentation with block-diagonal prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3818–3825, 2014.
- [10] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [11] Alvina Goh and René Vidal. Segmenting motions of different types by unsupervised manifold clustering. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–6. IEEE, 2007.
- [12] Amit Gruber and Yair Weiss. Multibody factorization with uncertainty and missing data using the em algorithm. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–707. IEEE, 2004.
- [13] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.
- [14] Kuang-Chih Lee, Jeffrey Ho, and David J Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(5):684–698, 2005.
- [15] Chun-Guang Li, Chong You, and René Vidal. Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework. *IEEE Transactions on Image Processing*, 26(6):2988–3001, 2017.
- [16] Zhouchen Lin, Minming Chen, Leqin Wu, and Yi Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Technical report, UIUC Technical Report, 2009.
- [17] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):171–184, 2013.
- [18] Yi Ma, Allen Y Yang, Harm Derksen, and Robert Fossum. Estimation of subspace arrangements with applications in modeling and segmenting mixed data. *SIAM review*, 50(3):413–458, 2008.
- [19] Jean-Marc Odobez and Patrick Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of visual communication and image representation*, 6(4):348–365, 1995.
- [20] Omar Oreifej and Mubarak Shah. *Robust Subspace Estimation Using Low-Rank Optimization*. Springer, 2014.
- [21] Shankar Rao, Roberto Tron, Rene Vidal, and Yi Ma. Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(10):1832–1845, 2010.
- [22] Wei Siming and Lin Zhouchen. Analysis and improvement of low rank representation for subspace segmentation. *arXiv preprint arXiv:1107.1561*, 2011.
- [23] Yasuyuki Sugaya and Kenichi Kanatani. Multi-stage unsupervised learning for multi-body motion segmentation. *IEICE Transactions on Information and Systems*, 87(7):1935–1942, 2004.
- [24] Roberto Tron and René Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.
- [25] René Vidal, Roberto Tron, and Richard Hartley. Multiframe motion segmentation with missing data using powerfactorization and gpca. *International Journal of Computer Vision*, 79(1):85–105, 2008.
- [26] Andrew Wagner, John Wright, Arvind Ganesh, Zihan Zhou, Hossein Mobahi, and Yi Ma. Toward a practical face recognition system: Robust alignment and illumination by sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(2):372–386, 2012.
- [27] Shandong Wu, Omar Oreifej, and Mubarak Shah. Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1419–1426. IEEE, 2011.
- [28] Huan Xu, Constantine Caramanis, and Sujay Sanghavi. Robust PCA via outlier pursuit. *IEEE Transactions on Information Theory*, 58(5):3047–3064, 2012.
- [29] Jingyu Yan and Marc Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Computer Vision—ECCV 2006*, pages 94–106. Springer, 2006.
- [30] Teng Zhang, Arthur Szlam, Yi Wang, and Gilad Lerman. Hybrid linear modeling via local best-fit flats. *International journal of computer vision*, 100(3):217–240, 2012.