Distortion Impact on Low-Dimensional Manifold Recovery of High-Dimensional Data

Alaa E. Abdel-Hakim
Electrical Engineering Department
Assiut University
Assiut, Egypt
alaa.hakim@ieee.org

Motaz El-Saban
Microsoft Research
Cairo Innovation Lab
Cairo, Egypt
motazel@microsoft.com

***Abstract*—In this paper, we investigate the effect of various kinds of distortions on the performance of subspace recovery using Principal Component Analysis (PCA). We verify that while PCA demonstrates relative good stability characteristics in presence of mild distortions, it suffers from major shortcomings to gross corruption of input observations, even if these corruptions are sparse. We present a performance evaluation study for the performance of the classical PCA under observation distortions and compare it with the performance of the Robust Principal Component Analysis (RPCA) approach. We verify the effectiveness of RPCA in lower-dimensional subspace recovery under different kinds of distortions.**

I. Introduction

Inference of low-dimensional structure of high-dimensional data has drawn much attention during the last years. The rapid increase of applications, where data usually lie in dimensions up to six-digit figures, mandates exploiting subspace recovery techniques. Principal Component Analysis (PCA) is considered as one of the well-known successful tools from this aspect [3]. The role of PCA is growing in several fields, e.g. image processing, computer vision, object recognition, information retrieval, audio/video processing, bioinformatics, and web search.

However, presence of noise and distortion affects the calculated principal components and hence the overall performance of PCA. The degradation of PCA performance depends on the nature of existing distortions, which vary from added small noise to large corruption of the input data [2]. For example, in image processing, an input image may be distorted by mild additive noise resulting from the acquisition device and can go up to an occlusion of a considerable part of the image. While small-noise distortions have relatively small impact on the estimated principal components, gross distortions, even sparse, may ruin the entire operation of PCA [3], [5]. So, the need for a distortion-robust method for recovery of low-dimensional manifolds of higher-dimensional data arises.

Some approaches have been developed to "robustify" the classical PCA. One of these approaches is the Robust Principal Component Analysis (RPCA) [5]. RPCA and low-rank matrix recovery have been used in several applications, e.g. matrix completion [2], face recognition [6], photometric stereo [7], compressed sensing [8]. RPCA solves the problem of recovering low-dimensional structure of high-dimensional data. It exploits the classical PCA assumption that an input data matrix is constructed by modifying a low-rank matrix, whose rank is much smaller than the dimension of the input data, by a sparse error term. RPCA recovers the subspace representation using a convex optimization model. Existence of erroneous and distorted data is very common in most of acquisition systems. So, in this paper, we evaluate the performance of both PCA and RPCA in presence of various types of distortions in input data.

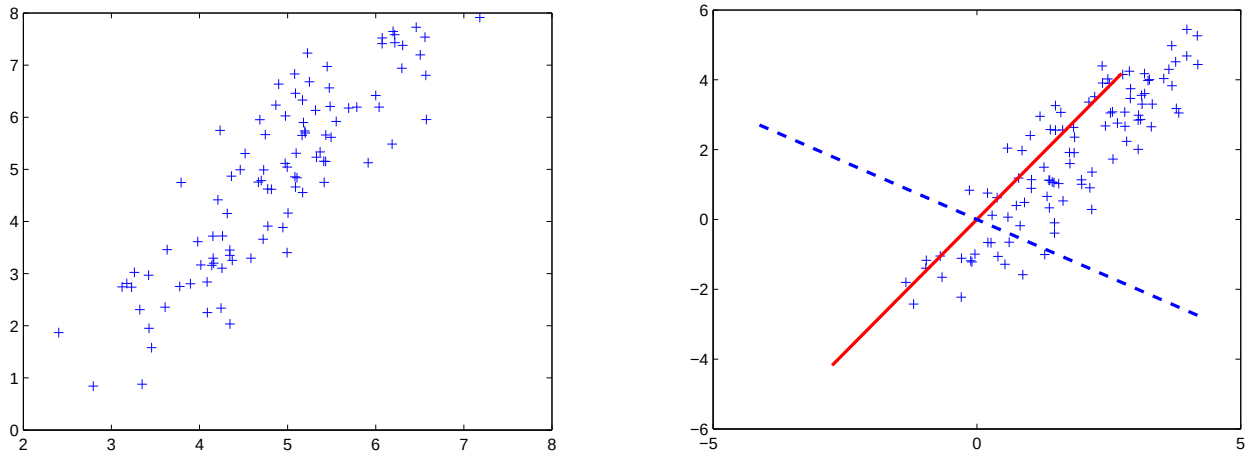
The rest of this paper is organized as follows: in section two, the principle operation of PCA is reviewed. The RPCA is explained in section three. The impact of distortion on both of them is discussed in section four. The design of experiments is shown in section five. Evaluation results are shown in section six. Finally, section seven concludes the paper.

II. Principal Component Analysis

In this section, we show the principle of operation of PCA and how it is applied to infer the low-dimensional structure of high-dimensional data.

Assume that we have a set of input observations, $D \in \mathbb{R}^{m \times n}$, where n is the number of instances in the input observation dataset and m is the dimensions of input data, i.e. the number of the features measured of the input data. As stated in the introduction, the key assumption under which PCA works is that the input data has a low rank, i.e. m is redundantly large. Hence, a lower dimension manifold of the input observations needs to be extracted. Figure 1 illustrates the concept of the subspace recovery problem using a 2D example. As shown in the figure, although the input data lies in a 2D space, it swings around a lower one-dimensional manifold. In other words, we can consider that the data is a low-rank matrix whose rank is smaller than its dimension.

PCA infers the lower-dimensional manifold by exploiting the eigenvectors of the covariance matrix of the feature-mean-adjusted input data, which correspond to the highest eigenvalues. For instance, the largest eigenvalue of the data shown in Fig 1 is much larger than the smaller eigenvalue. In other words, most of the information content of the input data is concentrated along the eigenvector(s) (which is stamped as principal component(s)) that correspond(s) to the largest eigenvalue(s).



(a) Original data

(b) The calculated principal components with the mean-adjusted data

Fig. 1. Operation of PCA on a sample of 2D dataset. Notice how the principal component, which corresponds to the highest eigenvalue (shown in the solid red line, is aligned to the major axis that the data is distributed along.

The following algorithm shows how the principle components are calculated [3].

Algorithm 1 PCA Algorithm

1: Consider input data $D \in \mathbb{R}^{m \times n}$, which is arranged such that every instance-observation is represented in a row-vector whose every element is a feature value.
2: Calculate the mean-adjusted data,D_{adj}, by subtracting every row of the input observation from the mean-feature vector.
3: Calculate the covariance matrix of D_{adj}, Σ.
4: Find the eigenvalues and the eigenvectors of Σ.
5: The principal components are the eigenvectors whose largest eigenvalues. The number of the selected principal components depends on the desired reduction of dimensionality.
6: Linear transformation to the new feature space is performed by multiplying the principal components by the adjusted data instances.

III. Robust Principal Component Analysis

RPCA handles the problem of inferring low-dimensional structure of high-dimensional data in a different way. It assumes that high-dimensional observed data, $D \in \mathbb{R}^{m \times n}$, is composed of a low rank term, $A \in \mathbb{R}^{m \times n}$, and an additive error corruption term, $E \in \mathbb{R}^{m \times n}$. The RPCA problem is formulated as follows [5]:

RPCA Problem:

Given $D = A + E \in \mathbb{R}^{m \times n}$, where A is unknown low rank and E is unknown sparse, recover A.

Traditional PCA tries to solve this problem via estimating the optimal A that satisfies the following constrained optimization [4]:

$$\min_{A,E} \parallel E \parallel_F \; s.t. rank(A) \leq r, D = A + E \tag{1}$$

where $\parallel . \parallel_F$ is the Frobenius norm. Computing the Singular Value Decomposition (SVD) of D optimally estimates A. The solution is estimated by projecting the columns of D onto the subspace spanned by the principal left singular vectors of D [4], [3]. Estimation of A using this procedures suffers from gross errors and distortions, even if they are sparse. So, Wright et.al. [5] showed that low-rank matrix A can be recovered from $D = A + E$ by solving the optimization problem shown in Eq. 2.

$$\begin{aligned} (A, E) \quad = \quad & \arg_{A,E} \min rank(A) + \lambda \parallel E \parallel_0 \\ & s.t. D = A + E \end{aligned} \tag{2}$$

The optimization model shown in Eq. 2 is not easy to be solved from an algorithmic prospective as it is highly nonconvex problem and contains two NP-hard terms [1], [5]. It has been shown in [5] that this problem can be solved by acceptably modifying the optimization model of Eq. 2 into a convex optimization problem as shown in Eq. 3.

$$\begin{aligned} (A, E) \quad = \quad & \arg_{A,E} \min \parallel A \parallel_* + \lambda \parallel E \parallel_1 \\ & s.t. D = A + E \end{aligned} \tag{3}$$

where $\parallel . \parallel_*$ and $\parallel . \parallel_1$ are the nuclear and L1-norms of a matrix, respectively, and λ is an arbitrary constant.

IV. Distortion Impact

In this section, we discuss the impact of distortion in the input observations on the performance of both PCA and RPCA.

For PCA, the major impact is represented in having gross error/distortion, which causes the calculated principal components to be deviated from their original position. For example, in Fig. 1(b), if instances, even few, of the input observations are grossly distorted, the calculated principal components will largely move from its original positions. This is resulted from the fact that this type of gross distortion, even sparse, affects both the calculated mean-feature vector, the calculated eigenvalues, and eigenvectors.

This is not the case of RPCA. As seen in Eq. 3, the convex optimization model optimizes for both of the minimum rank of A and the sparsest E. So, existence of gross, yet sparse, distortion will have minimum impact on the recovered low-rank matrix, since it is extracted away via the optimization process. This is from which the robustness of the RPCA comes.

V. Experimental Setup

In this section, we explain the design of the experiments that are used to evaluate the performance of both PCA and RPCA under various kinds of distortions, which vary from mild to gross corruption. We opted to use image data in our experiments. We use the known cameraman benchmark image, shown in Fig 2, as the ground truth or the "clean" observation. Distorted observations are generated by corrupting the clean image using five different types of distortions with various levels. In the next subsection, we illustrate each type of these five distortions. A distorted instance of the ground truth is generated by distorting the ground truth by a specific distortion type with a specific distortion level. An input observation dataset is built up by a set of distorted instances per each type of distortion and per each distortion level. The size of the input observation datasets in this experiments is set to 100 instances per dataset. Every instance represents a distorted image rearranged in a row format.

The performance of both PCA and RPCA is quantified by measuring the deviation of the recovered data from the ground truth. It is worthy mentioning that all principal components in PCA are used for recovery. This is made on purpose to assure that the measured recovery error is caused by the distortions of the input data only without any interference of the approximation error that is usually resulted by dimensionality-reduction. Equations 4 and 5 show the used formulae for recovery error calculations.

$$e_{PCA} = \frac{\| D_{PCA} - I * M \|_F}{\| I * M \|_F} \tag{4}$$

where I is the ground truth image stacked in a column vector and M is a unit row-vector $\in \mathbb{R}^{1\times m}$. $m = w \times h$, where w and h are the width and height of the input image, respectively.

Fig. 2. The noise-free ground truth image that is used in the evaluation experiments.

$$e_{RPCA} = \frac{\| A - I * M \|_F}{\| I * M \|_F} \tag{5}$$

where A is the recovered low-rank matrix using RPCA.

A. Distortions Used in the Experiments

1) Uniform distortion: In this type of distortion, a uniform random distortion is added to the intensity value of the distorted pixels. The added distortion is calculated by multiplying the maximum intensity value by a random value that is uniformly distributed in the range of $\pm L$, where L is the distortion level.

2) Salt & pepper distortion: Intensity values of 0 and 255 are added to the intensity value of randomly-selected distorted pixels. The distortion level L represents the percentage of distorted pixels in the image.

3) Gaussian distortion: A Gaussian random noise is added to the intensity value of the distorted pixels. The added noise has a zero-mean and a variable variance, which is referred to by the distortion distortion level, L.

4) Multiplicative distortion: A multiplicative noise is added to the distorted pixels. The added noise is proportional to the pixel intensity and equals to the intensity value multiplied by a uniformly-distributed random variable between 0 and the distortion level L.

5) Occlusion distortion: In this type of distortion, a white-pixel occlusion is applied to a random region of the input image of area equals to the distortion level multiplied by the area of the input image. The occluding box is truncated if it is positioned such that a part of it is located outside the boundaries of the image.

Figures 3 and 4 show exemplar images under these kinds of distortions with sample distortion levels.

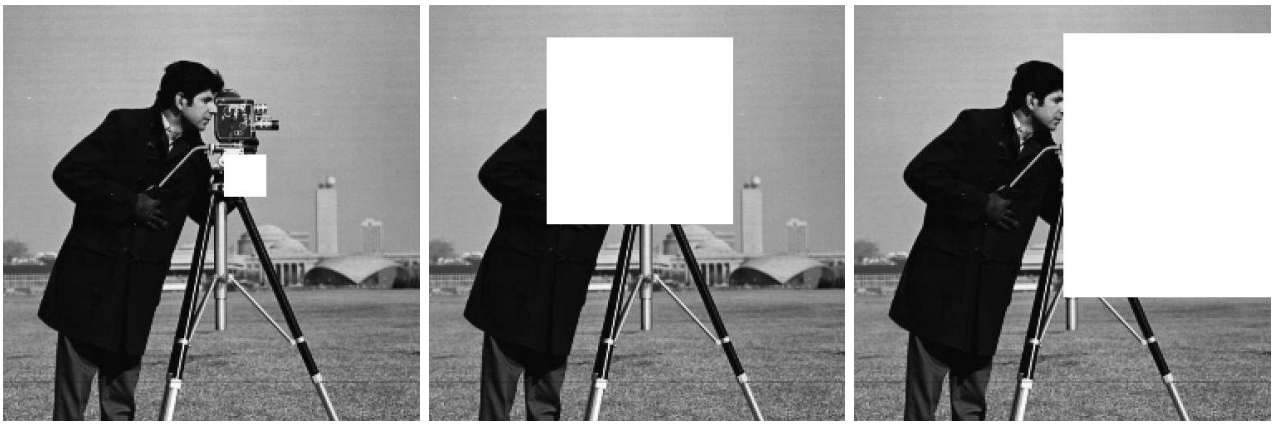
(a) Uniform distortion: L = 1% (b) Uniform distortion: L = 20% (c) Uniform distortion: L = 40%

(d) Salt & pepper distortion: L = 1% (e) Salt & pepper distortion: L = 20% (f) Salt & pepper distortion: L = 40%

(g) Gaussian distortion: L = 1% (h) Gaussian distortion: L = 20% (i) Gaussian distortion: L = 40%

Fig. 3. Exemplar images after being subjected to uniform, salt & pepper, and Gaussian distortions with different distortion levels.

(a) Multiplicative distortion: L = 1% (b) Multiplicative distortion: L = 20% (c) Multiplicative distortion: L = 40%



(d) Occlusion distortion: L = 1% (e) Occlusion distortion: L = 20% (f) Occlusion distortion: L = 40%

Fig. 4. Exemplar images after being subjected to multiplicative and occlusion distortions with different distortion levels.

VI. Evaluation Results

In this section, we show the obtained evaluation results for both PCA and RPCA under the above-mentioned kinds of distortions. Figures 5-9 show the recovery error curves for both PCA and RPCA under these kinds of distortions. It is noticed that the performance of RPCA is always better than that of PCA. In [5], the recommended distortion level for the operation of RPCA is below 30%. However, as we see in the figures, although the distortion levels go up to 40%, the RPCA still outperforms PCA despite the increase of its recovery error. The oscillations in the error curves of Fig. 9 are resulted from the truncation of the occluding boxes, when they are partially-positioned outside the image boarders. In such cases, the distortion levels are usually less than the predetermined ones.

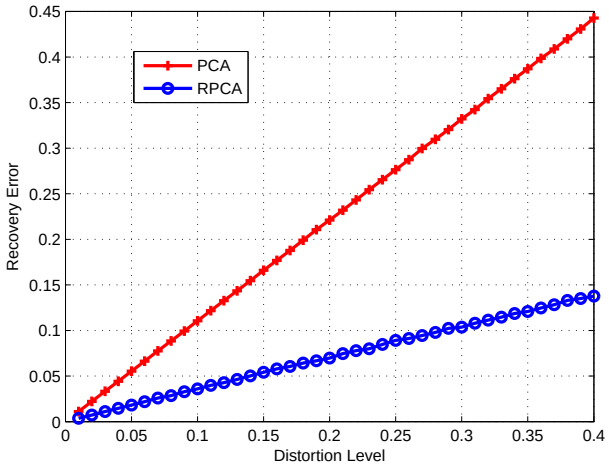


Fig. 5. The recovery error for both PCA and RPCA under different levels of uniform distortion.

VII. Conclusion

We presented a comparative performance evaluation study of both PCA and RPCA. The purpose of this study is to

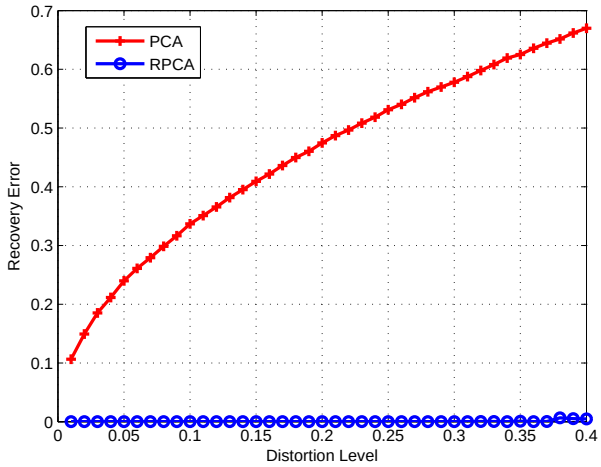


Fig. 6. The recovery error for both PCA and RPCA under different levels of salt & pepper distortion.

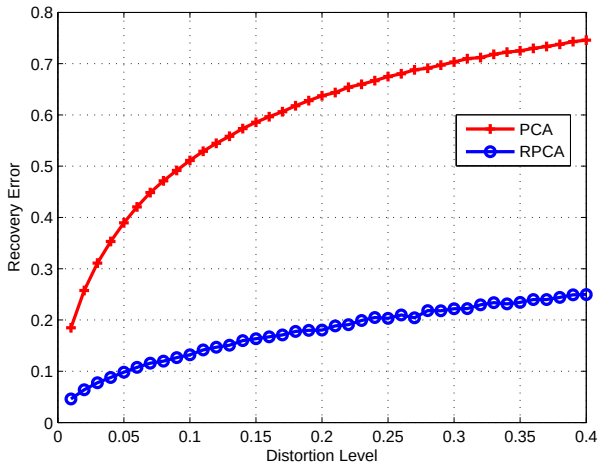


Fig. 7. The recovery error for both PCA and RPCA under different levels of Gaussian distortion.

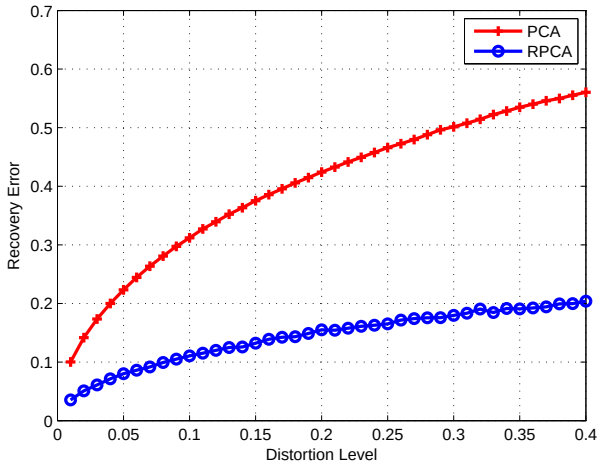


Fig. 8. The recovery error for both PCA and RPCA under different levels of multiplicative distortion.

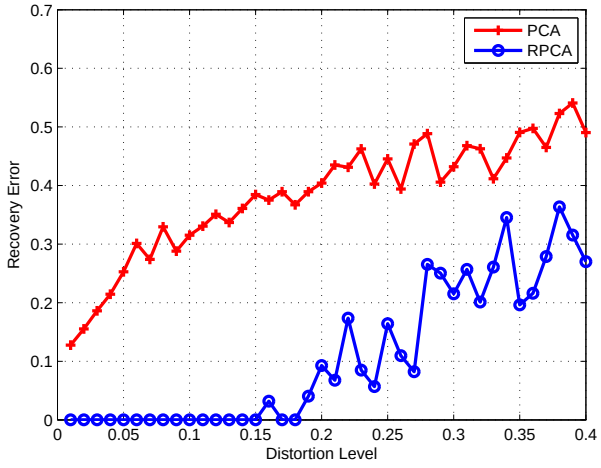


Fig. 9. The recovery error for both PCA and RPCA under different levels of occlusion distortion.

investigate the performance of both approaches under various kinds and levels of distortions. RPCA succeeds in recovering lower-dimensional subspace manifolds of higher-dimensional data under gross corruption, with acceptable error ranges. RPCA outperforms the classical PCA for all tested distortions. These results verify the effectiveness of the RPCA's convex optimization model in robust recovery of lower-dimensional manifolds of higher-dimensional data.

REFERENCES

[1] E. Amaldi and V. Kann. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209(1–2):237–260, 1998.

[2] E. J. Candes and Y. Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

[3] I. Jolliffe. *Principal Component Analysis*. Springer Verlag, New York, New York, 1986.

[4] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–18, 2009.

[5] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. In *proceedings of Neural Information Processing Systems (NIPS)*, December 2009.

[6] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31:210–227, February 2009.

[7] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma. Robust photometric stereo via low-rank matrix completion and recovery. In *Proceedings of the 10th Asian conference on Computer vision - Volume Part III*, ACCV'10, pages 703–717, Berlin, Heidelberg, 2011. Springer-Verlag.

[8] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman Iterative Algorithms for L1-Minimization with Applications to Compressed Sensing. *SIAM J. Imaging Sciences*, 1(1):143–168, 2008.