2014 22nd International Conference on Pattern Recognition

Covariance-Based PCA for Multi-Size Data

Menghua Zhai, Feiyu Shi, Drew Duncan, and Nathan Jacobs
Department of Computer Science, University of Kentucky, USA
{mzh234, fsh224, drew, jacobs}@cs.uky.edu

***Abstract*—Principal component analysis (PCA) is used in diverse settings for dimensionality reduction. If data elements are all the same size, there are many approaches to estimating the PCA decomposition of the dataset. However, many datasets contain elements of different sizes that must be coerced into a fixed size before analysis. Such approaches introduce errors into the resulting PCA decomposition. We introduce CO-MPCA, a nonlinear method of directly estimating the PCA decomposition from datasets with elements of different sizes. We compare our method with two baseline approaches on three datasets: a synthetic vector dataset, a synthetic image dataset, and a real dataset of color histograms extracted from surveillance video. We provide quantitative and qualitative evidence that using CO-MPCA gives a more accurate estimate of the PCA basis.**

I. INTRODUCTION

For many natural datasets the elements can be accurately represented as a linear combination of a small number of basis vectors. For a given dataset, the optimal basis vectors, and their corresponding coefficients, can be computed using principal component analysis (PCA). This popular matrix factorization is used for numerous applications ranging from face recognition [12] to outdoor-scene appearance modeling [4]. In these applications, the PCA decomposition is used as a low-dimensional representation. Methods for estimating more accurate representations in these domains can improve the accuracy of face recognition methods and scene factorization methods for Internet photo collections.

Computational tools for computing the PCA decomposition of a matrix, such as the singular value decomposition (SVD) or eigenvector analysis, assume that the input data can be represented as fixed-length vectors. Many datasets can be represented as vectors, for example image datasets can be *vectorized* by converting each $w \times h$ RGB image into a $3wh \times 1$ vector. However, for many real-world datasets the elements come in a variety of shapes, sizes and resolutions. The standard approach to this problem is to convert each element to the same size. In image datasets, this is performed by cropping or a resizing the images prior to *vectorizing*. These operations result in a loss of information and introduce artifacts into the resultant PCA decomposition.

We propose a maximum likelihood (ML) approach to the multi-size PCA problem. We extend the covariance-based approach to estimating a PCA decomposition. Instead of making all of the vectors a fixed size and then computing a covariance matrix, we directly estimate the covariance matrix from the multi-sized data using nonlinear optimization. We compare our method with two baseline approaches: UP-MPCA, which first converts all vectors to a fixed size, and FD-MPCA, which uses a frequency domain representation. Based on an evaluation with three datasets, we find that our method gives more accurate estimates of the PCA decomposition, and hence could be used to improve the results for many applications that rely on PCA to find low-dimensional data representations. We see significant increases in accuracy as the dataset size increases and as the range of possible data vector sizes increases.

A. Related Work

Our method is closely related to work in two domains.

a) Joint Subspace Estimation and Image Alignment: Image motion can significantly increase the rank of an image dataset and, therefore, significantly reduce the interpretability and value of the PCA decomposition. For example, in the case of pure rotational motion the PCA decomposition is equivalent to the full-rank discrete cosine transform (DCT) [13] regardless of the pattern. To address the problem of image motion, numerous methods have been proposed for simultaneously aligning a set of images and estimating a low-rank basis. Jepson et al. [7] propose an online approach for estimating a low-rank appearance model for object tracking. De la Torre and Black [2] focus on face image datasets and solve this problem using nonlinear optimization. Peng et al. [10] propose a method for aligning linearly correlated images, with sparse errors, using a series of convex optimization problems. A basic assumption in these methods is that the motion is unknown but small, and that the images are of a similar size. In this work, we focus on more general datasets and assume that the transformation between data vectors, which could be images, are known in advance. We focus on the problem of minimizing artifacts introduced by differences in data vector sizes.

b) Image Super-Resolution: The objective of super-resolution is to combine multiple low-resolution images into a higher-resolution image [9], [15]. Methods for this problem include both frequency and spatial domain approaches [9], although spatial-domain approaches are more popular due to their ability to more naturally incorporate realistic observation models. Several methods for incorporating subspace constraints have been used. Capel and Zisserman [1] proposed to reconstruct the higher resolution image using a PCA basis estimated from a training set of high-resolution images. They constrain an ML solution to lie on the subspace, or use the subspace to define priors for a MAP estimate. Unlike this work, we do not assume an independent set of high-resolution images to learn a subspace. Vandewalle et al. [14] propose to solve the super-resolution problem using a subspace method, but their focus is on solving the image alignment problem for super-resolution. We assume the transformations are known in advance and directly estimate the PCA decomposition from a mixture of low and high-dimensional data vectors.

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.284

II. Background: PCA for Fixed-Size Data

Principal component analysis (PCA) estimates a low-dimensional representation of an input dataset, along with a corresponding basis [8]. Let $\mathbf{X} = \{X_i | i = 1 \dots N\}$ denote the input dataset, with $X_i \in \mathbb{R}^{d_i \times 1}$. If we assume that each data vector is the same size, namely $d_i = d$, the dataset, $\mathbf{X} = [X_1, X_2, \dots, X_N]$, can be represented as a $d \times N$ matrix. The goal of PCA is to compute a p-dimensional basis, $\mathbf{\Phi} \in \mathbb{R}^{d \times p}$, and a corresponding set of coefficients, $\mathbf{h} = [h_1, h_2, \dots, h_N]$ where $h_i \in \mathbb{R}^{p \times 1}$, that can be used to accurately reconstruct the input dataset.

Our description of PCA assumes that input data elements can be represented by vectors. This is a minor restriction because multi-dimensional input data elements, such as images, can usually be converted into a vector representation. For instance, an $m \times n$ image with three color channels can be represented by a $d \times 1$ vector, where $d = m \times n \times 3$. Throughout the paper, we will assume that all multi-dimensional input data has been "vectorized" using a similar method.

A. Covariance Matrix

A standard approach for estimating the principal components is based on the eigenvalue decomposition of the data covariance matrix. The covariance matrix, $\mathbf{C}$, of the data is defined as:

$$\mathbf{C} = \frac{1}{N-1} \tilde{\mathbf{X}} \tilde{\mathbf{X}}^\mathsf{T} \tag{1}$$

where $\tilde{\mathbf{X}} = [X_1 - \bar{X}, X_2 - \bar{X}, \dots, X_N - \bar{X}]$ is the mean-centered data matrix. We then decompose the covariance matrix, $\mathbf{C}$, to obtain the PCA basis, $\mathbf{\Phi}$, as follows:

$$\mathbf{C} = \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Phi}^\mathsf{T} \tag{2}$$

where $\mathbf{\Phi}$ is the set of eigenvectors, which are guaranteed to be orthogonal, and $\mathbf{\Sigma}$ is a diagonal matrix whose non-zero entries are the eigenvalues. The optimal estimate of the PCA coefficients, $\mathbf{h}$, is $\mathbf{\Phi}^\mathsf{T} \tilde{\mathbf{X}}$.

B. Singular Value Decomposition

When the dimension of data is large, computing its covariance matrix is time-consuming and memory-intensive. The singular value decomposition (SVD) method eliminates the need to construct a covariance matrix and is more efficient in such cases. To estimate PCA using this method, we apply SVD to the centered data matrix, $\tilde{\mathbf{X}}$:

$$\tilde{\mathbf{X}} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^\mathsf{T}. \tag{3}$$

To see how the PCA basis is computed from the SVD method, we relate it to the covariance method. By inserting Equation (3) into Equation (1), we obtain the following relation:

$$\mathbf{C} \propto \tilde{\mathbf{X}} \tilde{\mathbf{X}}^\mathsf{T} = (\mathbf{U} \mathbf{\Lambda} \mathbf{V}^\mathsf{T})(\mathbf{U} \mathbf{\Lambda} \mathbf{V}^\mathsf{T})^\mathsf{T} = \mathbf{U} \mathbf{\Lambda}^2 \mathbf{U}^\mathsf{T}. \tag{4}$$

Comparing this with Equation (2), it follows that $\mathbf{\Phi} = \mathbf{U}$ and $\mathbf{\Sigma} \propto \mathbf{\Lambda}^2$, so the column vectors of $\mathbf{U}$ are the desired basis. As with the covariance case, we can now easily estimate $\mathbf{h}$.

III. PCA for Multi-Size Data

Datasets with different sizes are common outside of laboratory settings. There are many possible causes of this including upgrading to a new sensor with higher resolution, changing the binning strategy in histogram estimation, and inherent differences in scale, for example the image patches of objects at different distances from a camera. When data vectors, X_i, are different sizes, the data cannot be represented as a matrix and neither of the methods for computing PCA described in the previous section work as formulated.

Our approach to solving this problem combines ideas from work on image super-resolution with the covariance-based method for estimating a PCA decomposition. We begin by defining a generative model for multi-size data.

A. Generative Model for Multi-Size Data

In image super-resolution, low-resolution images are presumed to be generated from a high-resolution image using a transformation model [1], [3], [5]. A typical transformation model is described as follows: $X_i = D_i B_i W_i Y + n_i$, for $i = 1, \dots, N$, where Y is a single high-resolution image, X_i is one of many low-resolution images, n_i is a noise term and D_i, B_i, W_i, are decimation, blurring and warping matrices, respectively. The super-resolution problem is to simultaneously estimate each transformation model and obtain the high-resolution image.

We assume that the transformation models are known and that there are multiple high-resolution images (otherwise estimating a PCA basis is not particularly useful). This gives us the following model:

$$X_i = S_i Y_i, \text{ for } i = 1, \dots, N, \tag{5}$$

where Y_i denotes a high-resolution data vector and S_i is the known transformation model. Since we assume we are attempting to estimate a single high-resolution PCA basis, the transformation matrices, $\{S_i\}$, are $m_i \times n$ matrices with $m_i < n$. Although a linear transformation operator, S_i, seems quite simple, it can model a wide variety of image transformations, including planar image transformations [11].

Our goal is to estimate a p-dimensional PCA basis, $\mathbf{\Phi}$, of the full-sized dataset, $\mathbf{Y}$. We first describe two baseline approaches, and then propose our multi-size covariance method.

B. Upsampling Approach (UP-MPCA)

When the data transformation is a scaling operation, a simple solution to this problem is to upsample all data vectors in $\mathbf{X}$ to the full-resolution. There are many techniques for upsampling, all of which introduce artifacts. In this work, we use bilinear interpolation and write the upsampling operation as $S_i^\dagger X_i$. We use a standard PCA algorithm on the upsampled data vectors to obtain a high-resolution decomposition. We first estimate the average data vector, $\bar{Y}$, as follows:

$$\bar{Y} \approx \frac{1}{N} \sum_{i=1}^{N} S_i^\dagger X_i. \tag{6}$$

We then subtract $\bar{Y}$ from each data vector, $\{X_i\}$, and compute the covariance matrix, C.

Because information is lost when downsampling, $S_i Y_i$, the inverse transformation, $S_i^\dagger$, is imperfect. In general, the transformation from low-resolution, back to a high-resolution image introduces blur. This blur in the individual data vectors is then reflected in the average image, $\bar{Y}$, and the covariance matrix, C. Performing PCA on the resulting covariance matrix leads to artifacts in the decomposition. This approach performs well when the inverse of the transformation operation, $S_i^\dagger$, is easy to estimate, such as when image scale changes are relatively small. However, it fails dramatically with large changes in scale.

C. Discrete Cosine Transform Approach (FD-MPCA)

In this approach, we convert the various data vectors into the frequency domain using the discrete cosine transform (DCT). If we restrict consideration to transformations, S_i, that only change the scale of the data, we can treat higher-frequencies that are not observed in the smaller data vectors as missing values. We then form these vectors into a matrix and use an off-the-shelf approach for solving for PCA basis that handles missing values [6]. Since the DCT is an invertible linear operation, we can easily convert the frequency-domain PCA basis into a spatial-domain basis. This approach, though somewhat limited in that it is only suitable for a restricted class of data transformations, has the advantage that it is able to use fast existing solvers.

IV. Covariance-Based PCA for Multi-Size Data

The two baseline methods described above have significant shortcomings: The upsampling method, UP-MPCA, depends on the ability to accurately invert the data transformations, S_i. Unfortunately, this is not possible, for example with large reductions in scale. The frequency domain method, FD-MPCA, depends on the missing values in the frequency domain being isolated to a small number of entries. This depends significantly on the type of transformation that occurs.

We propose CO-MPCA, a method that overcomes these issues by using maximum-likelihood estimation (MLE) to estimate the covariance matrix, C. Our overall approach is described in Algorithm 1. We formulate two objective functions to avoid the need to approximate the inverse of the data transformations, $S_i^\dagger$. The first objective function is for the problem of centering the data vectors prior to estimating the covariance matrix. This requires an estimate of the dataset average, which is non-trivial because the data vectors are of different sizes. We describe our approach to this subproblem, which is essentially a super-resolution algorithm, in the following section.

Algorithm 1: CO-MPCA

Input: Multi-size dataset, $\mathbf{X}$
Output: PCA basis $\mathbf{\Phi}$

1. Compute the average vector, $\bar{Y}$, using MLE.
2. Compute the centered data vectors, $\{\tilde{X}_i = X_i - S_i\bar{Y}\}$.
3. Estimate an initial covariance matrix, $C^0 = cov(\{S_i^\dagger X_i\})$, using independently upsampled data vectors.
4. Find the covariance matrix, $\tilde{C}$, that minimizes $E(\tilde{C})$ using nonlinear optimization (PCG).
5. Estimate the PCA basis, $\mathbf{\Phi}$, using SVD (Equation (4)).

A. Multi-Size Average Image Estimation

As defined in Equation (1), the first step in estimating the covariance matrix is centering the data vectors. This is trivial when the vectors are the same size, but more challenging with multi-size data. The approach defined in Equation (6) fails when the inverse of S_i is difficult to approximate. We avoid this problem by following an approach from image super-resolution. We use MLE to estimate the average vector, $\bar{Y}$, as follows: $\bar{Y} = \arg\min_{\bar{Y}} E(\bar{Y})$, where [1]

$$E(\bar{Y}) = \frac{1}{N}\sum_{i=1}^{N} \left\| S_i\bar{Y} - X_i \right\|_2^2 = \frac{1}{N}\sum_{i=1}^{N} (S_i\bar{Y} - X_i)^\intercal (S_i\bar{Y} - X_i). \quad (7)$$

We optimize $E(\bar{Y})$ using standard nonlinear optimization methods together with the following analytical gradient:

$$\frac{\partial E(\bar{Y})}{\partial \bar{Y}} = \frac{2}{N}\sum_{i=1}^{N} S_i^\intercal (S_i\bar{Y} - X_i). \quad (8)$$

This approach avoids the problem of directly inverting each data vector using $S_i^\dagger$ and gives more accurate estimates of the average image, $\bar{Y}$. We define the centered data vectors as follows, $\tilde{X}_i = X_i - S_i\bar{Y}$.

B. Multi-Size Covariance Estimation

Given a set of centered multi-size data vectors, $\{\tilde{X}_i\}$, we use MLE to estimate the covariance matrix. Since the covariance matrix is the average of the rank-1 outer products of the input data, we propose the following objective function to estimate the full-size covariance matrix:

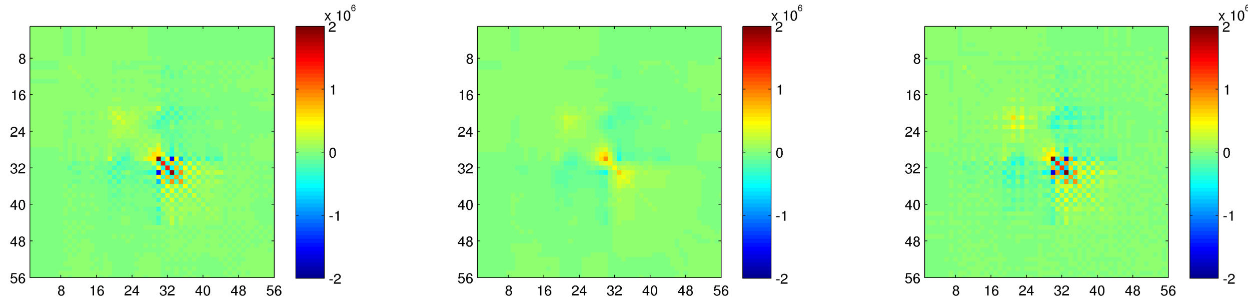
$$E(\tilde{C}) = \sum_{i=1}^{N} \left\| S_i\tilde{C}S_i^\intercal - \tilde{X}_i\tilde{X}_i^\intercal \right\|_F^2 = \sum_{i=1}^{N} \mathbf{tr}\left\{ \left(S_i\tilde{C}S_i^\intercal - \tilde{X}_i\tilde{X}_i^\intercal\right)^\intercal \left(S_i\tilde{C}S_i^\intercal - \tilde{X}_i\tilde{X}_i^\intercal\right) \right\}. \quad (9)$$

We optimize this objective function using preconditioned nonlinear conjugate gradient descent (PCG) with the following analytical gradient:

$$\frac{\partial E(\tilde{C})}{\partial \tilde{C}} \propto \sum_{i=1}^{N} S_i^\intercal S_i\tilde{C}S_i^\intercal S_i - S_i^\intercal \tilde{X}_i\tilde{X}_i^\intercal S_i. \quad (10)$$

While there are many choices for nonlinear optimization, we find that, in practice, PCG converges more quickly than the alternatives for this problem. As a starting condition, we estimate an initial covariance matrix, $C^0 = cov(\{S_i^\dagger X_i\})$, using independently upsampled data vectors.

[1] $\|\cdot\|_2$ denotes Euclidean norm and $\|\cdot\|_F$ denotes Frobenius norm.



(a) Ground-Truth (Empirical)

(b) UP-MPCA

(c) CO-MPCA

Fig. 1. A comparison of covariance matrices that shows that our proposed method (CO-MPCA) generates a more accurate estimate (right) of the ground-truth covariance (left) than the upsampling method (middle).

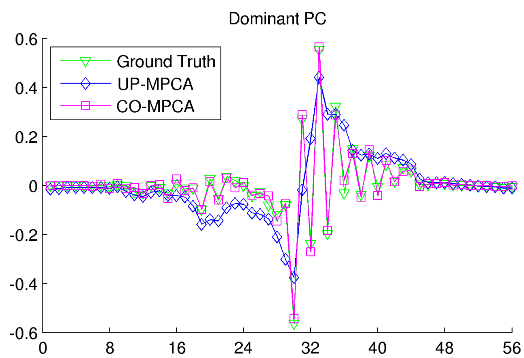


Fig. 2. The dominant principal component extracted from each of the covariance matrices in Figure 1.

Figure 1 compares the full-size covariance matrix estimated by CO-MPCA to that estimated by the UP-MPCA (using data from Section V-C) applied to multi-sized data. We see that the covariance estimate from CO-MPCA is much closer to the ground-truth covariance than the covariance estimated from UP-MPCA. In addition, Figure 2 shows the resulting first principal components from each of the three methods. This highlights the fact that more accurate covariance estimates lead to more accurate principal component estimates.

V. Evaluation

We evaluate our proposed method, and the two baseline methods, on three datasets: a 1D synthetic data, 2D synthetic data and one derived from image histograms. For each dataset, we generate the full-size dataset and then randomly resize each full-size data vector to obtain the multi-size dataset. For all experiments, the ground-truth PCA decomposition is computed from the full-size data.

We define two error metrics that evaluate the quantitative performance of our method. The first metric measures the difference between the full-size empirical principal components and those estimated using one of our multi-size PCA methods:

$$e_{pc} = \left\| |\mathbf{\Phi}_{gt}^{\mathsf{T}} \mathbf{\Phi}_{ms}| - \mathbf{I} \right\|_F^2 , \quad (11)$$

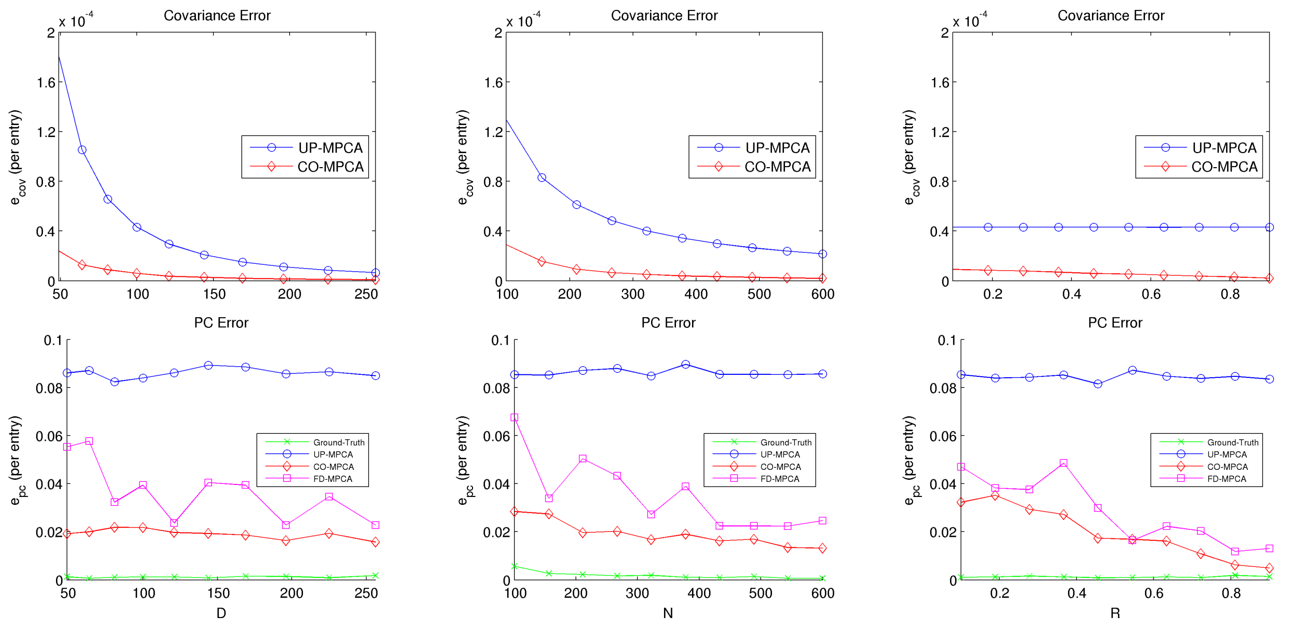
where $\mathbf{\Phi}_{gt}$ is the set of principal components derived from the full-size dataset and $\mathbf{\Phi}_{ms}$ is the result of a multi-size PCA method applied to the multi-size dataset. If $\mathbf{\Phi}_{ms} = \mathbf{\Phi}_{gt}$ then the error is zero. Our second error metric directly compares the covariance matrix of the upsampling method and the MLE covariance-estimation approach. This error metric is defined as follows:

$$e_{cov} = \|C_{em} - C_{ms}\|_F^2 , \quad (12)$$

where C_{em} is deduced directly from the raw full image dataset $\mathbf{Y}$. This metric does not apply to to the DCT-based approach because that method does not require an estimate of the covariance matrix.

A. 1D Synthetic Data

We first test the multi-size PCA methods, on a set of randomly generated 1D datasets. We compare the performance of all methods under differing dimensions, D, dataset size, N, and amounts of resizing, R. For each dataset, we first generate a full-size basis, $\mathbf{\Phi}_{gt}$, by orthogonalizing a $D \times D$ matrix with entry values drawn uniformly at random from $[0, 1]$. We then generate a data matrix as follows, $\mathbf{Y} = \mathbf{\Phi}_{gt} diag([1^\gamma, 2^\gamma, 3^\gamma, \ldots]) randn(P, N)$, where $\gamma = -.4$, $diag(\mathbf{v})$ and $randn(m, n)$ represent for, respectively, diagonal matrix with the elements of vector $\mathbf{v}$ on the main diagonal and $m \times n$ matrix that each element is randomly selected with standard normal distribution. To generate the multi-size data, we resize each column vector by sampling a resizing ratio, r, uniformly at random in the range of $[R, 1]$, where a resizing ratio of 1 means no resizing (we use a box-shaped kernel for vector resizing). We generate datasets based on these parameter settings. Unless otherwise specified, the dimensionality $D = 100$, the number of vectors $N = 300$, and the low bound of resizing ratio $R = 0.5$. The results of this experiment are shown in Figure 3. The covariance matrix estimated using CO-MPCA is closer to the full-size empirical covariance matrix than the one estimated by UP-MPCA. In the same figure we also show the accuracy of the resulting PCA basis. Here we see that CO-MPCA outperforms both the UP-MPCA method and the FD-MPCA method.



(a) dimension (b) dataset size (c) resize range

Fig. 3. The first row compares the covariance errors of CO-MPCA and UP-MPCA. The second row shows the PC reconstruction errors among different methods; the empirical method applies the standard PCA method on the full size images, which is closest to ground-truth. All these plots imply that the CO-MPCA outperforms the other two multi-size PCA methods in reducing reconstruction errors.

B. 2D Synthetic Data

Similar to the 1D synthetic data, we also compared each method using synthetic image data. Our ground-truth images are 15×15 and were generated in a similar fashion as in the previous section. The key difference is that when we make the smaller size version of the dataset we first convert the data vectors to a 2D array, resize (with a lower-bound size of 7×7), and then convert back into a vector. The multi-size images are resized from the ground-truth dataset with box-shaped kernel. This experiment highlights the flexibility of the linear transformation model, S_i, because it shows that it can represent both 1D and 2D scaling operations. Figure 4 shows that when the number of images increase, our method outperforms the upsampling method on both of our metrics.

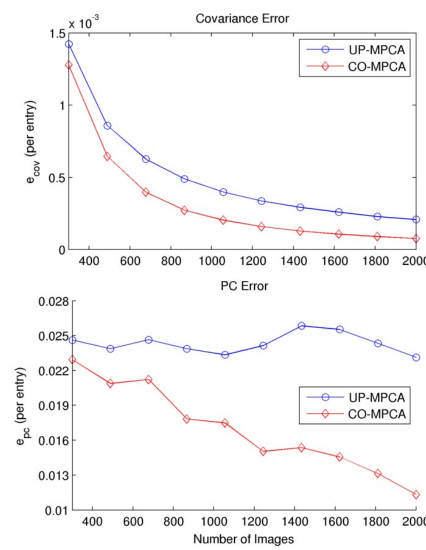


Fig. 4. A comparison of covariance error and PC error for 2D synthetic data. The CO-MPCA method gives more accurate covariance matrices and decompositions, especially as image sizes increase.

C. Color Histogram

As a final evaluation, we use a collection of color histograms of images from an urban traffic scene video(Figure 5). All the images were captured by a static camera. To generate multi-size data in this scenario, we extract color histograms with different numbers of color bins.

Suppose an n-bin histogram $\mathbf{h} = (h_1, h_2, ..., h_n)$ is to be resized to an m-bin histogram $\mathbf{g} = (g_1, g_2, ..., g_m),\ m < n$. Let $\mathbf{H}$ and $\mathbf{G}$ denote the accumulative functions of $\mathbf{h}$ and $\mathbf{g}$ respectively. We expect:

1) $G_m = H_n$
2) $H_{\lfloor i' \rfloor} \leq G_i \leq H_{\lceil i' \rceil},\quad i' = \frac{i \times n}{m}$

A natural definition of G_i that satisfies the principles is:

$$G_i = H_{\lfloor i' \rfloor} + \frac{i - \lfloor i' \rfloor}{\lceil i' \rceil - \lfloor i' \rfloor}(H_{\lceil i' \rceil} - H_{\lfloor i' \rfloor}). \tag{13}$$

Given the accumulative function $\mathbf{G}$, the corresponding histogram $\mathbf{g}$ is easily computed.

TABLE I. PERFORMANCE COMPARISON OF UP-MPCA AND CO-MPCA ON COLOR HISTOGRAM DATA.

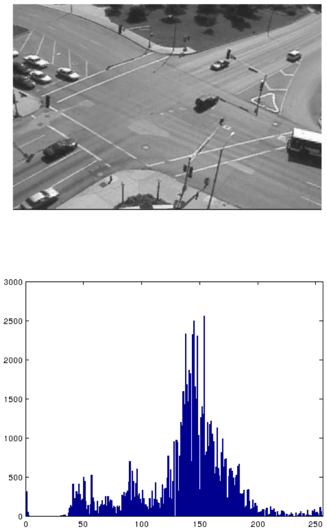
Color Scale		PC Reconstruction Error (e_{pc})	
Min	**Max**	**UP-MPCA**	**CO-MPCA**
48	96	2.4180	1.1427
71	142	2.2764	1.2513
82	164	2.6330	1.2419
105	210	1.8396	1.0326
128	256	2.2349	1.4477

Table I shows the results of an experimental comparison of UP-MPCA and CO-MPCA using a selection of 200 images from the video. We generate a multi-size dataset by computing a color histogram with the number of bins ranging uniformly at random between an upper and lower bound for each image. We compute the reconstruction error of the first 5 principal component vectors under different ranges of color scales. The results show that the CO-MPCA is roughly twice as accurate as the UP-MPCA method in this case.

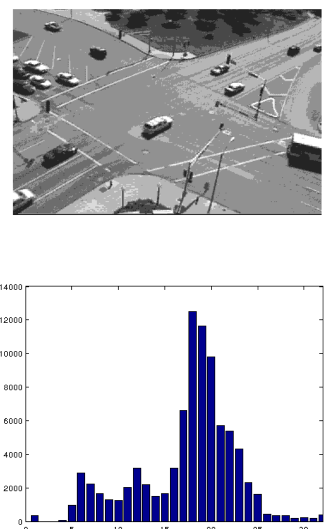
VI. CONCLUSIONS

We proposed and evaluated several algorithms for computing a PCA decomposition of datasets with different sizes. We defined a new method for estimating PCA decompositions that uses nonlinear optimization to estimate the data covariance matrix in such cases. We show quantitatively that this method is superior to two baseline approaches on synthetic and real datasets. Our method is ideal for cases in which a large number of data vectors exist, and we require high-accuracy in estimating the PCA decomposition.

Our work has two main limitations. First, for high-dimensional data the minimization process will be very memory and CPU intensive due to the size of the covariance matrix, $\tilde{C}$. This is a fundamental problem with covariance-based methods for estimating a PCA decomposition. Second, we assume that the data transformations are linear operations. Handling high-dimensional data and nonlinear transformations are both interesting areas for future work.



(a) 256 grayscale bins



(b) 32 grayscale bins

Fig. 5. Example images from our traffic scene dataset and corresponding color histograms computed at different bin resolutions.

REFERENCES

[1] D. Capel and A. Zisserman. Super-resolution from multiple views using learnt image models. In *CVPR*, 2001.

[2] F. De la Torre and M. J. Black. Robust parameterized component analysis. In *ECCV*, 2006.

[3] M. Elad and A. Feuer. Super-resolution reconstruction of image sequences. *IEEE Trans. on PAMI*, 1999.

[4] R. Garg, H. Du, S. M. Seitz, and N. Snavely. The dimensionality of scene appearance. In *ICCV*, 2009.

[5] B. Gunturk, A. Batur, Y. Altunbasak, M. Hayes, and R. Mersereau. Eigenface-domain super-resolution for face recognition. *IEEE Trans. on Image Processing*, 2003.

[6] A. Ilin and T. Raiko. Practical approaches to principal component analysis in the presence of missing values. *The Journal of Machine Learning Research*, 99:1957–2000, 2010.

[7] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust online appearance models for visual tracking. *IEEE Trans. on PAMI*, 2003.

[8] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2005.

[9] S. C. Park, M. K. Park, and M. G. Kang. Super-resolution image reconstruction: a technical overview. *Signal Processing Magazine, IEEE*, 2003.

[10] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma. Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. In *CVPR*, 2010.

[11] S. M. Seitz and S. Baker. Filter flow. In *ICCV*, 2009.

[12] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 1991.

[13] M. Uenohara and T. Kanade. Optimal approximation of uniformly rotated images: Relationship between karhunen-loeve expansion and discrete cosine transform. *IEEE Trans. on Image Processing*, 1998.

[14] P. Vandewalle, L. Sbaiz, J. Vandewalle, and M. Vetterli. Super-resolution from unregistered and totally aliased signals using subspace methods. *IEEE Trans. on Signal Processing*, 2007.

[15] J. Yang and T. Huang. Image super-resolution: Historical overview and future challenges. In *Super-resolution imaging*. CRC Press, 2010.