Published in proceedings of ÖAGM'02, pages 71-78, 2002.

Incremental approach to robust learning of eigenspaces [1)]

Danijel Skočaj and Aleš Leonardis
University of Ljubljana, Slovenia
{danijels,alesl}@fri.uni-lj.si, http://www.cogvis.fri.uni-lj.si

Abstract:
The standard PCA approach to visual learning of representations is intrinsically non-robust and usually performed in a batch mode, which is inadmissible in a real-world on-line scenario. In this paper we propose a novel method for robust and incremental learning of eigenspaces. The method sequentially updates the representation using the previously acquired knowledge for determining consistencies and discarding inconsistencies in the input images. We show the experimental results, which demonstrate the advantages and disadvantages of the proposed approach.

1 Introduction

Learning of object and scene representations is an essential part of any cognitive vision system. In the real world learning is usually a continuous, never-ending process, thus requiring incremental methods for updating previously learnt representations. In addition, one can also expect that the images of objects and scenes are not always ideal and as such they may contain noise or occlusions, and therefore require a robust learning algorithm. Often, visual learning is approached as appearance-based modeling of objects and scenes, which is commonly realized using Principal component analysis (PCA). However, the standard PCA approach is usually performed in the batch mode, i.e., all training images are processed simultaneously, which means that all of them have to be given in advance. This is inadmissible in an on-line scenario, where the images to be processed are obtained sequentially. We propose a novel method for incremental learning, which processes images sequentially one by one and updates the principal subspace accordingly. We also embed the method for robust recognition [4] in this incremental framework, which results in an overall method for incremental robust learning.

The proposed algorithm presents a new approach to robustness in the appearance-based modeling using PCA. A number of methods for robust recognition have already been proposed (see e.g., [4]). A few methods for robust learning have been introduced as well [2, 6], but all

[1)] This work was supported in part by the EU project *CogVis (IST-2000-29375)* and the grants funded by the Ministry for Education, Science, and Sport: Research Program *Computer Vision-1539-506* and *SLO-A/07*.

of them operate in a batch mode. Several algorithms for incremental learning have also been proposed [3, 1], however, they are not robust. Our method merges these two learning approaches by sequentially updating the representation using the previously acquired knowledge for determining consistencies and discarding inconsistencies in the input images.

The paper is organized as follows. First, we briefly outline the standard PCA approach. Then we propose a method for incremental learning. In section 4 we present a robust version of this method. After that we present the experimental results. Finally, we summarize the paper and outline some work in progress.

2 Standard PCA

First, we briefly outline the standard PCA approach and introduce the notation. Let $\mathbf{x}_i = [x_{1i}, \ldots x_{Mi}]^\top \in \mathbb{R}^M$ be an individual image represented as a vector and $\mathbf{X} = [\mathbf{x}_1, \ldots \mathbf{x}_N] \in \mathbb{R}^{M \times N}$ the matrix of all training images. For further processing, we create the mean normalized data matrix $\hat{\mathbf{X}}$ by subtracting the mean image $\boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i$ from the input images. Principal axes are traditionally obtained by performing the eigendecomposition (or, similarly, the singular value decomposition) of the covariance matrix of the input data. They are denoted by $\mathbf{u}_i = [u_{1i}, \ldots u_{Mi}]^\top \in \mathbb{R}^M$; $\mathbf{U} = [\mathbf{u}_1, \ldots \mathbf{u}_N] \in \mathbb{R}^{M \times N}$. The columns of $\mathbf{U}$, i.e., the eigenvectors, are arranged in a decreasing order with respect to the corresponding eigenvalues. Usually, only k, $k < N$, eigenvectors with the largest eigenvalues are needed to represent $\hat{\mathbf{x}}$ to a sufficient degree of accuracy as a linear combination of eigenvectors: $\tilde{\mathbf{x}} = \sum_{i=1}^{k} a_i \mathbf{u}_i = \mathbf{U}\mathbf{a}$, where $\tilde{\mathbf{x}}$ denotes the approximation of $\hat{\mathbf{x}}$. A coefficient vector characterizing an image is traditionally calculated by a standard projection of the input image into the eigenspace: $\mathbf{a} = \mathbf{U}^\top \hat{\mathbf{x}}$.

3 Incremental PCA

In this section we propose a novel method for incremental learning. It takes the training images sequentially and computes the new eigenspace from the subspace obtained in the previous step and the current input image.

Let us suppose that we have already built an eigenspace from the first n images. In the step $n + 1$ we could calculate a new eigenspace from the *reconstructions* of the first n input images and a new image using the standard batch method. The computational complexity of such an algorithm would be prohibitive, since at each step we would have to perform the batch PCA on a set of high-dimensional data. However, identical results can be obtained by using low-dimensional *coefficients* of the first n input images instead of their high-dimensional reconstructions, since coefficients and reconstructed images encompass the same visual variability, i.e., they are the same points represented in different coordinate frames. Since the dimension of the eigenspace is very small, this algorithm is computationally very efficient.

The summarized algorithm for updating an eigenspace looks as follows:

Input: current mean value $\boldsymbol{\mu}^{(n)}$, current eigenvectors $\mathbf{U}^{(n)}$, current coefficients $\mathbf{A}^{(n)}$, new input image $\mathbf{x}$. [1)]

Output: new mean value $\boldsymbol{\mu}^{(n+1)}$, new eigenvectors $\mathbf{U}^{(n+1)}$, new coefficients $\mathbf{A}^{(n+1)}$, new eigenvalues $\mathbf{l}^{(n+1)}$.

Algorithm:

1: Project a new image $\mathbf{x}$ in the current eigenspace: $\mathbf{a} = \mathbf{U}^{(n)\top}(\mathbf{x} - \boldsymbol{\mu}^{(n)})$.
2: Reconstruct the new image: $\mathbf{y} = \mathbf{U}^{(n)}\mathbf{a} + \boldsymbol{\mu}^{(n)}$.
3: Compute the residuum vector: $\mathbf{r} = \mathbf{x} - \mathbf{y}$. $\mathbf{r}$ is orthogonal to $\mathbf{U}^{(n)}$.
4: Append $\mathbf{r}$ as a new basis vector: $\mathbf{U}' = \left[\begin{array}{cc} \mathbf{U}^{(n)} & \frac{\mathbf{r}}{||\mathbf{r}||} \end{array} \right]$.
5: Determine the coordinates of the coefficients in the new basis: $\mathbf{A}' = \left[\begin{array}{cc} \mathbf{A}^{(n)} & \mathbf{a} \\ \mathbf{0} & ||\mathbf{r}|| \end{array} \right]$.
6: Perform PCA on $\mathbf{A}'$. Obtain the mean value $\boldsymbol{\mu}''$, the eigenvectors $\mathbf{E}''$ and eigenvalues $\mathbf{l}''$.
7: Optionally drop the least significant vector of the new basis: $k^{(n+1)} = k^{(n)}$,
$\mathbf{U}'' = [\mathbf{e}''_1, \dots \mathbf{e}''_{k^{(n+1)}}]$.[2)]
8: Project the coefficients to the new basis: $\mathbf{A}^{(n+1)} = \mathbf{U}''^{\top}(\mathbf{A}' - \boldsymbol{\mu}''\mathbf{1}_{1\times n+1})$.[3)]
9: Rotate the subspace $\mathbf{U}'$ for $\mathbf{U}''$: $\mathbf{U}^{(n+1)} = \mathbf{U}'\mathbf{U}''$.
10: Update the mean: $\boldsymbol{\mu}^{(n+1)} = \boldsymbol{\mu}^{(n)} + \mathbf{U}'\boldsymbol{\mu}''$.
11: New eigenvalues: $\mathbf{l}^{(n+1)} = [l''_1, \dots l''_{k^{(n+1)}}]$.

The initial values of the mean image, the eigenvectors, and the coefficients can be obtained by applying the batch PCA on a small set of images or simply by setting the first training image for the initial eigenspace by assigning $\boldsymbol{\mu}^{(1)} = \mathbf{x}_1$, $\mathbf{U}^{(1)} = \mathbf{0}_{M\times 1}$, and $\mathbf{A}^{(1)} = 0$.

It is worth noting that this algorithm estimates the identical principal subspace as the method proposed by Hall et al. [3]. However, the subspace is obtained in *a different way.* A significant advantage of our method is that it is able to treat different images differently, which enables to advance it into a weighted incremental method [5]. Furthermore, our method maintains the low dimensional representations of the previously learnt images throughout the entire learning stage, meaning that each training image can be discarded immediately after the update.

We will demonstrate the behavior of the proposed algorithm on a simple 2-D example. The 2-D input space contains 41 points shown as black dots in Fig. 1(a). The goal is to estimate 1-D principal subspace, i.e., the principal axis. The eigenspace is being built incrementally.

[1)] Superscript denotes the step which the data is related to. $\mathbf{U}^{(n)}$ denotes the values of $\mathbf{U}$ at the step n.
[2)] By discarding a basis vector we preserve the dimension of the eigenspace.
[3)] $\mathbf{1}_{m\times n}$ denotes a matrix of dimension $m \times n$, where every element equals to 1.

At each step one point (from the left to the right) is added to the representation and the eigenspace is updated accordingly. Fig. 1(a) illustrates how the eigenspace evolves during this process. The principal axis, obtained at every sixth step, is depicted. The points, which were appended to the model at these steps, are marked with crosses. One can observe, how the origin of the eigenspace (depicted as a square) and the orientation of the principal axis change through time, adapting to the new points, which come into the process. In the end, the estimated eigenspace, which encompasses all training points, is almost equal to the eigenspace obtained using the batch method.

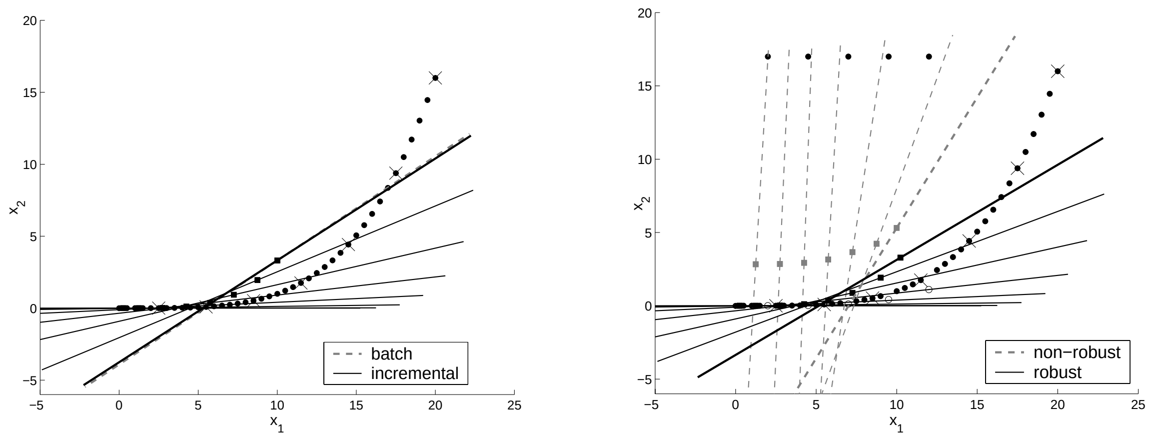
4 Robust approach

The incremental algorithm presented in the previous section suffers from similar drawbacks as the standard batch algorithm, including non-robustness to the non-gaussian noise. Here we show how to increase its robustness.

In the robust framework we are aware that images may contain outliers. We treat as outliers all pixels, which are not consistent with the information contained in other images. Since at each step we have a current model of the object or scene seen so far, we can detect outliers in the new image and replace them with the values, which are yielded by the current model. This is achieved by projecting the new image into the current eigenspace in a robust manner. Instead of a simple projection, a robust procedure based on subsampling and hypothesize-and-select paradigm is used [4]. Coefficients are obtained mainly from inliers, thus their reconstructions tend to the correct values in outliers as well. Consequently, the reconstruction error in outliers is high, which makes their detection easier. Therefore, to make the incremental learning robust, we first detect outliers in a new image, reconstruct their values and update the eigenspace with restored, outlier-free image using the algorithm described in the previous section.

To demonstrate the behavior of the the robust algorithm, we significantly changed the values of the second coordinate of five points in our 2-D example. Fig. 1(b) shows that these outlying points pull the origin and the orientation of the estimated principal axis in the wrong direction. On the other hand, the robust method sequentially detects the outlying coordinate values, replaces these values with the reconstructed values (shown as circles) and updates the eigenspace accordingly. The principal axis obtained using this approach is significantly closer to the optimal one shown in Fig. 1(a).

A potential drawback of incremental methods is a propagation of errors, since images are added sequentially. In the case of the robust incremental algorithm this could be problematic if the initialization of the algorithm is not performed well. The initial eigenspace, which is obtained by using the batch method, should be reliable and stable. It should roughly model



(a) (b)

Figure 1: (a) Incremental learning. (b) Robust incremental learning.

heterogeneous views of an object or a scene. Therefore, the images from various parts of the image sequence should be used for initialization. When the model encompasses a sufficient number of views it becomes more stable and this is no longer a problem. Certainly, this initialization also has to be performed in a robust manner, using a robust batch algorithm [6].

5 Experimental results

We performed several experiments on synthetic and real data. Here, we present two sets of experiments, which demonstrate the advantages and disadvantages of the proposed method.

First, we exhibit the performance of the non-robust incremental method. We built eigenspaces of various dimensions from 720 images of twenty objects from the COIL database (Fig. 2(a)). Fig. 2(b), depicts the mean squared reconstruction errors (MSRE) of the images reconstructed from the coefficients obtained by projecting the training images into the eigenspaces, which were built using the batch method (in plots indicated as *batch*) and the proposed incremental method (*incXseq*). The results are very similar; MSRE obtained using the incremental method is only 3.1% worse on the average. The curve *incAseq* represents reconstruction errors of images obtained from the coefficients, which were calculated at that time instant, when the particular image was added to the model and then maintained throughout the process of incremental learning. Using this approach, an image can be discarded immediately after the model is updated. As one can observe, the squared reconstruction errors are still quite similar. In this case, the degradation of the results is 8.6% on the average. In this experiment the images were coming into the learning process in a sorted order, i.e., first all images of the first object, then all images of the second object, and so on. In the second experiment we changed this sequence by giving the training images to the learning process in random order. Thus, the eigenspace in the early stage of the learning process already encompassed images of several objects. Therefore, it was a good approximation of the final eigenspace. The incoming training images in the later stages were just refining the current eigenspace. Consequently, the results have improved. MSRE produced by *incArnd* and *incXrnd* approaches, are only 3.1% and 1.3% worse than the results of the batch method, respectively.

Fig. 2(c) shows the MSRE of all 720 images for the dimension of the eigenspace 50. One can observe, that the curves representing the incremental approach follow the curve produced by the batch method very closely without large deviations over the whole sequence of images. All the results clearly indicate that the incremental method is almost as effective as the batch.

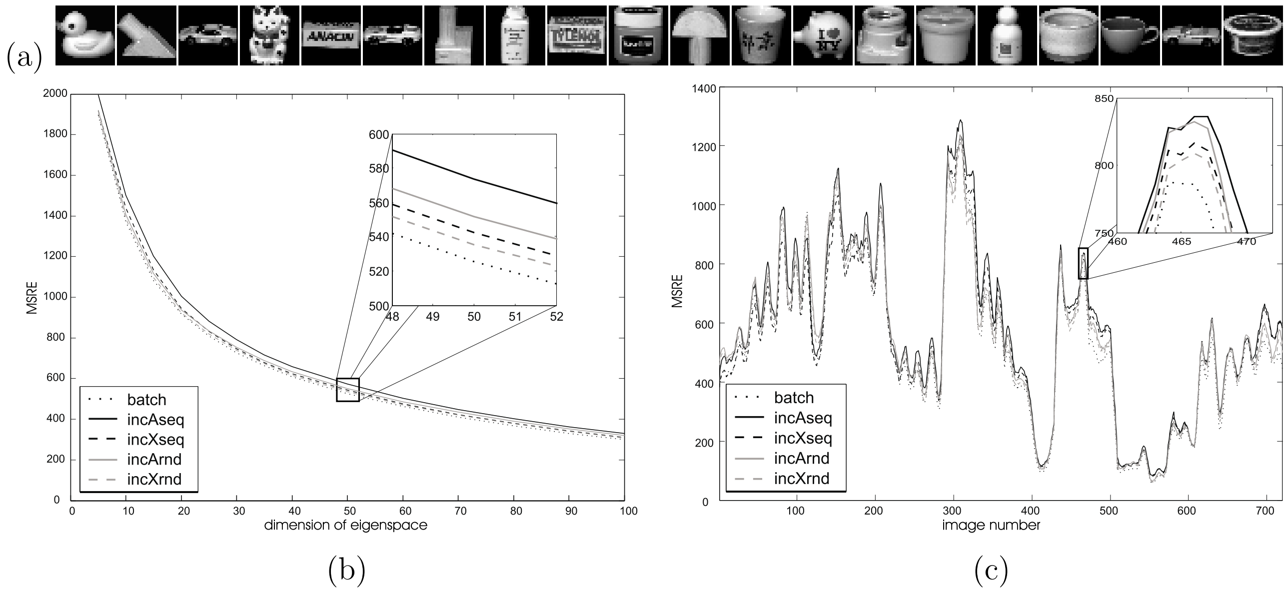


Figure 2: (a) Training images. MSRE produced by the batch and four versions of the incremental approach: (b) for various dimensions of the eigenspace, (c) for dimension 50.

We tested the performance of the proposed robust incremental method on the images with known ground truth. We synthetically applied gradual illumination changes and nonlinear illumination changes (a shadow—the vertical "cloud") to a set of 100 images. In addition, we added, as an outlier area, a square on a randomly chosen position in 80% of the images (see Fig. 3(a), the first row). The goal was to learn the representation capturing the illumination variations (linear and nonlinear) but discarding the outliers.

We tested several approaches to exhibit some properties of the proposed method. The results are given using two measures. The first measure is the mean squared reconstruction error of the reconstructed outlier-free (ground truth) images (Table 1). Besides MSRE, a precision/recall curve is given for each method in Fig. 3(b). A P/R curve indicates the utility of a method for outlier detection. If the learnt representation does not include outliers, this ability is high and the precision and recall values are close to 1. In addition, some reconstructed training images are visualized in Fig. 3(a) (rows two to five).

First, we applied the standard batch method on ground truth images, i.e., training images without outliers (*batchOnGT*), which produced optimal results. Then, we applied the standard batch method (*batchStd*) on the training images containing outliers, which generated poor results, since the standard method is sensitive to occlusions. Therefore, we applied the robust batch method [6] (*batchRob*), which produced better results. However, since a lot of occlusions were present in the training images, the results were still not satisfactory.

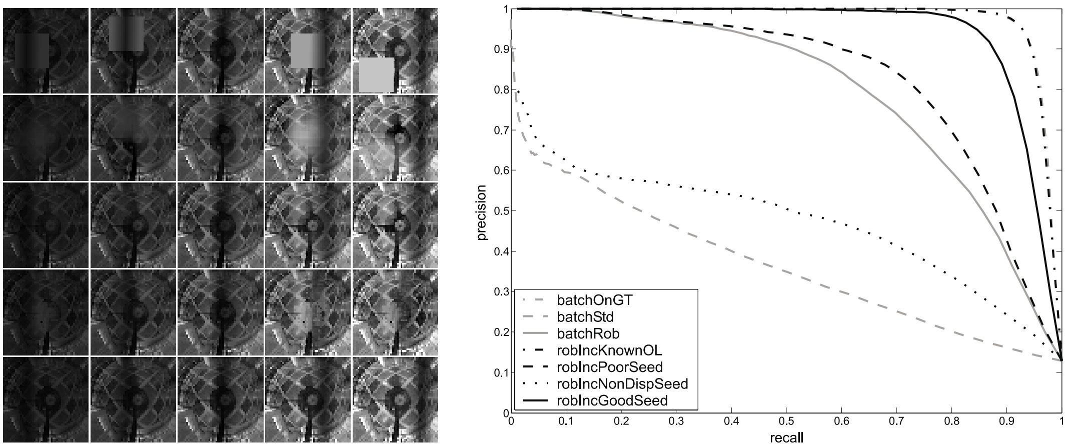
Then, we tested the proposed robust incremental method. First we applied this method under the assumption, that the occlusions were known (*robIncKnownOL*). Therefore, the algorithm did not have to detect outliers. The results are excellent; they are very close to the optimal ones. This means that the algorithm for updating the eigenspace works fine even if some data in the input images are missing and that the efficiency of the robust incremental algorithm mainly depends on the ability to detect outliers. It turns out that this ability significantly depends on the initial stage of the learning process. If the seed (the initial eigenspace, which is used for the initialization of the incremental algorithm) is not reliable and is affected by occlusions (*robIncPoorSeed*), the results are not very good. If the seed is too small and is built from the training images, which are not dispersed over the whole image sequence (*robIncNonDispSeed*), the results are very poor. To demonstrate this, we built the seed using a few images from the first half of the image sequence. Consequently, the first half of the images were reconstructed well, however the images from the end of the sequence were reconstructed badly. Since not even the rough appearance of these images was encompassed in the initial eigenspace, all the changes in these images were considered to be outliers and were not added into the representation. For this reason, the vertical cloud was not modelled correctly as can be observed in the fourth row of Fig. 3(a). At last, we built the seed from the images with the lowest reconstruction errors (images without outliers), which were evenly dispersed over the whole image sequence (*robIncGoodSeed*). This approach produced excellent results, which are rather close to the optimal ones. This indicates that when the eigenspace, which is being updated, is stable enough, i.e., roughly encompassing different views of objects or scenes, the outliers in the training images are successfully detected and correctly reconstructed.

batch			*robInc*			
OnGT	*Std*	*Rob*	*KnownOL*	*PoorSeed*	*NonDispSeed*	*GoodSeed*
1.7	61.1	29.8	2.0	21.2	166.0	2.9

Table 1: MSRE obtained using different learning methods and seeds.

6 Conclusion

In this paper we proposed a novel method for incremental learning of eigenspaces and advanced it into a robust incremental method. The method sequentially updates the representation using the previously acquired knowledge for determining consistencies and discarding inconsistencies in the input images.



(a) (b)

Figure 3: (a) From top to bottom: Training images, reconstructions using *batchStd*, *batchRob*, *robIncNonDispSeed*, and *robIncGoodSeed* approaches, respectively. (b) Precision/recall curves.

The experimental results show that in the case of outlier-free training images, the incremental method is almost as effective as the batch method. When the training images contain noise or occlusions, the performance of the robust learning algorithm depends on the early stages of the learning process. If the initial representation is built satisfactorily, the inconsistencies in other training images are successfully detected and the representation is robustly built, encompassing all consistent information in the training images and discarding outliers.

The proposed method is suitable for continuous on-line learning, where the model adapts to input images as they arrive (SLAM principle). The algorithm is flexible, since it is able to treat each image differently. Therefore, more recent (or more reliable, or more informative, or more noticeable) images can have a stronger influence on the model then others. The principles of short-term and long-term memory, forgetting, and re-learning can be implemented and investigated. This principles are the subject of the ongoing research.

References

[1] S. Chandrasekaran, B. S. Manjunath, Y. F. Wang, J. Winkeler, and H. Zhang. An eigenspace update algorithm for image analysis. *Graphical Models and Image Processing*, 59(5):321–332, September 1997.

[2] F. De la Torre and M. J. Black. Robust principal component analysis for computer vision. In *ICCV'01*, pages I: 362–369, 2001.

[3] P. Hall, D. Marshall, and R. Martin. Merging and spliting eigenspace models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):1042–1048, 2000.

[4] A. Leonardis and H. Bischof. Robust recognition using eigenimages. *Computer Vision and Image Understanding*, 78:99–118, 2000.

[5] D. Skočaj and A. Leonardis. Weighted incremental subspace learning. Submited, 2002.

[6] D. Skočaj, H. Bischof, and A. Leonardis. A robust PCA algorithm for building representations from panoramic images. In *Computer Vision – ECCV 2002*, volume IV, pages 761–775, May 2002.