Robust Incremental Linear Discriminant Analysis Learning by Autonomous Outlier Detection

Martina Uray[1], Horst Bischof[2] and Heinz Mayer[1]

[1] Institute of Digital Image Processing, Joanneum Research
{*martina.uray,heinz.mayer*} *@joanneum.at*

[2] Institute for Computer Graphics and Vision, Graz University of Technology
bischof@icg.tugraz.at

Abstract
Bringing robustness into subspace methods is very important, for training as well as for recognition. In case of Linear Discriminant Analysis (LDA) the task of robust classification is already solved, therefore, we focus on treating pixel outliers and occlusions in the training stage. More precisely, in this work we consider the task of incremental learning. Based on an augmented LDA basis that incorporates a certain amount of reconstructive information we are able to achieve the desired robustness. The advantage of the enriched basis is that it contains enough reconstructive information to handle noisy data, while it still exploits its full discriminative property.

1 Motivation

From a statistical point of view robustness of a method is defined as the property of being insensitive to outliers. A common measure for robustness is the breakdown point defining the amount of noise an estimator can handle. Thus, the higher the breakdown point the more robust is the method. In images noise usually appears due to non-optimal conditions during data acquisition (e.g., for live streams). In general, we define two types of outliers; global noise describes those images that do not fit to the dataset (*outlying images*) and local noise corresponds to non-Gaussian noise like missing pixels, occlusions, varying illumination or background (*outlying pixels*).

In the following we discuss the influence of local noise when working with subspaces. As a matter of fact, learning as well as recognition from noisy data is inaccurate. On the one hand, if trained from unreliable data, the subspace does not represent the classes well, while on the other hand, test data, which is too different from the learned representation cannot be classified correctly. Therefore, robust learning and robust recognition is essential. The term robustness for images, as it will be used in this work, is described by Fidler *et al.* [2] as the ability to detect outliers, consequently, work on uncorrupted pixels only (which means that only reliable information is used), resulting in a high breakdown point. Moreover, they presented a solution for the robust discriminative classification in the presence of pixel noise.

In contrast, we will handle the robust discriminative learning. Robust LDA training algorithms that can be found in literature deal with three issues. Li *et al.* [8] define a robust linear discriminant criterion based on the sample median and the median absolute derivation in order to handle wrong labeling of data. Other authors [6, 13] deal with the robust statistical estimation of the mean and the covariance matrix to achieve a good generalization, even if the number of training data is small. Identification and removal of atypical objects (outliers) was done by Roth [10] employing Kernel Fisher Discriminants. The main disadvantage of this method is that it only identifies images containing outliers in order to remove them. However, we want to discard only pixels and not complete images such that no useful information is lost. Thus, a reliable detection and appropriate handling of the outlying pixels is necessary. In addition, as far as we know the combination of robust (in terms of pixel noise) and incremental learning is nowhere tackled.

Figure 1 gives an example pointing out the need for a robust incremental learning method. We consider the construction of a subspace for two classes of faces. For clarity the plot is forced apart, such that each image in LDA space has a given distance to the others. Class centers are depicted as colored lines instead of just single points. The decision boundary between the two classes is drawn as black line. The starting basis consists of two undisturbed images of both persons, but each of the remaining training images is occluded by a black square encompassing 30% of its content (at a random position, Figure 1(a)). Test images in turn are noise-free (Figure 1(b)). It can be seen that, without a proper handling of the occlusions, the subspace is considerably affected by the corrupted updates. Consequently, not all test data can be classified correctly (Figure 1(c)). The robust updating on the other hand enables an appropriate separation between classes (Figure 1(d)).

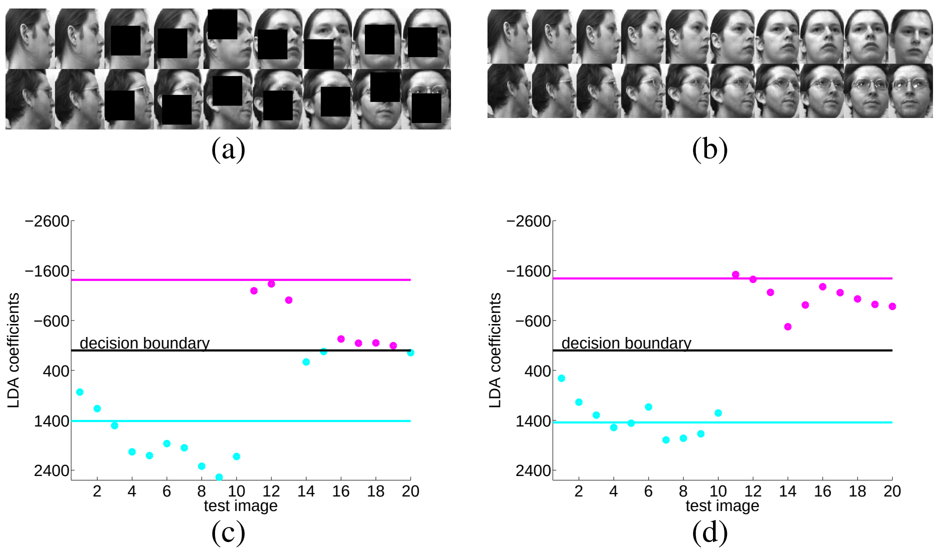


Figure 1: The advantage of robust LDA subspace updating. Even if starting from an undisturbed basis, the subspace adapts to the noisy data, resulting in a poor recognition. On the other hand, if the update is carried out in a robust way it represents only noise-free data giving the desired correct classification. (a) Training data, (b) Test data, (c) Non-robust results and (d) Robust results.

We distinguish between reconstructive and discriminative methods, both having certain advantages and disadvantages. Concerning robustness reconstructive methods can be applied in order to search for pixels that are not consistent with the data. This in turn is not possible for discriminative methods. In order to overcome this drawback Fidler *et al.* [2] have shown that it is possible to construct a discriminative subspace that contains additional reconstructive information, which allows for robust classification. In [15] this incorporated reconstructive information was already exploited to develop an efficient incremental subspace learning. The behavior we observed there resulted in the assumption that this type of basis allows for a robust extension of the updating as well (following the robust incremental Principal Component Analysis (PCA) learning developed by Skočaj [11]). We will show that due to the incorporation and exploitation of the reconstructive information of the augmented PCA basis (*aPCA* basis), it is possible to adapt these approaches to achieve robustness for the incremental LDA learning as well.

Summarizing, the advantage of the augmented basis is that on the one hand, it contains enough reconstructive information to enable a robust incremental training, while on the other hand, it keeps the full discriminative information for a reliable recognition. Furthermore, on the robustly built subspaces we can directly apply the robust classification, such that disturbed images are of little consequences in both stages. As it was seen in the incremental case from clean data [15] methods considering only one type of information have severe drawbacks.

For the following considerations, it is necessary to differentiate between two sets of indices. An image $\mathbf{x} \in \mathbb{R}^m$ is separated into its corrupted (missing pixels, MPL) and its uncorrupted (non-missing pixels) part. Indices of the missing pixels are defined as $\mathcal{I}^\circ = \{i \mid \mathbf{x}(i) = MPL\}$ and indices of the non-missing pixels are given as $\mathcal{I}^\bullet = \{i \mid i \notin \mathcal{I}^\circ\}$. Consequently, the combination of these indices $\mathcal{I}^\bullet \cup \mathcal{I}^\circ = \{i \mid i = 1, \ldots, m\}$ gives the complete image. All parameters marked with a bullet accord to the indices given in $\mathcal{I}^\bullet$ and all parameters marked with a circle accord to the indices given in $\mathcal{I}^\circ$.

2 Robust Incremental Updating by Subspace Combination

2.1 Subspace Methods

The goal of subspace methods is to find a projection that transforms the training data (images) such that new unknown images can be efficiently classified. For batch methods it is assumed that all training data is given in advance. Formally, given n images $\mathbf{x}_j \in \mathbb{R}^m$ represented as vectors which are organized in a matrix $\mathbf{X} = [\mathbf{x}_1, \ldots \mathbf{x}_n] \in \mathbb{R}^{m \times n}$. Let $\boldsymbol{\mu} \in \mathbb{R}^m$ be the mean image and let c be the total number of classes. In the following we take a closer look at the subspace methods that are relevant for our approach.

Principal Component Analysis [5] is an unsupervised method that estimates a low dimensional representation of the data that minimizes the squared reconstruction error. Therefore, the data $\mathbf{X}$ is projected onto a lower-dimensional subspace by

$$\mathbf{A} = \mathbf{U}^T \left(\mathbf{X} - \boldsymbol{\mu} \mathbf{1}_{1 \times n} \right) \ , \tag{1}$$

where the projection matrix $\mathbf{U}$ is built from the eigenvectors of the covariance matrix of $\mathbf{X}$ and $\mathbf{1}_{m \times n}$ denotes an $m \times n$ matrix of ones. Usually only k, $k \ll n \ll m$, columns are needed to reconstruct $\mathbf{X}$ to a desired accuracy:

$$\mathbf{X} \approx \mathbf{U}_k \mathbf{A}_k + \boldsymbol{\mu} \mathbf{1}_{1 \times n} \ , \tag{2}$$

where $\mathbf{U}_k = [\mathbf{u}_1, \ldots, \mathbf{u}_k] \in \mathbb{R}^{m \times k}$ is the truncated reconstructive basis and the rows of $\mathbf{A}_k = [\mathbf{a}_1^T, \ldots, \mathbf{a}_n^T]^T \in \mathbb{R}^{k \times n}$ are the image representations in the truncated subspace.

Linear Discriminant Analysis [3], in contrast, is a supervised method that also integrates the class label information of the training samples. It seeks for $(c-1)$ hyperplanes that are capable to separate the given data. This is realized by a projection $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_{(c-1)}] \in \mathbb{R}^{m \times (c-1)}$ that minimizes the intra-class scatter while it maximizes the inter-class scatter. Hence, a test sample $\mathbf{x}$ is projected onto the subspace by

$$g(\mathbf{x}) = \mathbf{W}^T (\mathbf{x} - \boldsymbol{\mu}) \ . \tag{3}$$

The class label l is then assigned according to the result of a nearest neighbor classification. Therefore, the Euclidean distances d of the projected sample $g(\mathbf{x})$ and the class centers $\boldsymbol{\nu}_i$ are compared:

$$l = \min_{i=1,\dots,c} d(g(\mathbf{x}), \boldsymbol{\nu}_i) \; . \tag{4}$$

2.2 Robust Incremental LDA Learning

Incremental subspace methods allow to directly add new images to an already built lower-dimensional representation. Thus, the original images can be discarded directly after they have been processed. In the following we will deduce a robust incremental update scheme for LDA that is based on the properties of augmented subspaces (keeping the complete discriminative information) originally developed for robust LDA classification [2]. Therefore, we first create an *aPCA* subspace by augmenting the k dimensional reconstructive subspace with additional $c-1$ vectors containing discriminative information. Those supplementary vectors are constructed from vectors that would be discarded when truncating the subspace to k-dimensions. In this way, the full discriminative information is maintained. Second, we build the actual LDA representation from the thus obtained augmented subspace. In this work we suppose that the starting basis is built from clean data. This simplifies the problem, but since our focus lies on the presentation of a robust update strategy this precondition is appropriate (for considerations about a robust LDA basis construction see [14]).

Assuming that a subspace representation was already built from n images the current augmented PCA subspace is given by the *aPCA* vectors $\mathbf{U}^{(n)} \in \mathbb{R}^{m\times(k+c-1)}$, the *aPCA* coefficients $\mathbf{A}^{(n)} \in \mathbb{R}^{(k+c-1)\times n}$, and the mean vector $\boldsymbol{\mu}^{(n)} \in \mathbb{R}^{m\times 1}$; the current augmented LDA representation is given by the vectors $\mathbf{V}^{(n)} \in \mathbb{R}^{(k+c-1)\times(c-1)}$ and the class centers $\boldsymbol{\nu}_i^{(n)} \in \mathbb{R}^{(c-1)\times 1}$.

To correctly update the given subspace with a noisy image $\mathbf{x}^{(n+1)} \in \mathbb{R}^m$ the outliers have to be detected and properly handled (Algorithm 1). We will show that the desired robustness can be achieved by application of the hypothesize-and-test paradigm, originally developed for robust PCA recognition [7, 1]. Due to the incorporated reconstructive part of the *aPCA* basis, this idea can be applied to detect outliers as those pixels that are not consistent with the current representation (see also [12]).

In the hypothesize step p points are randomly subsampled from the image and are assumed to be inliers. Using these points $\mathbf{x}^{(n+1)\bullet} \in \mathbb{R}^p$ an overdetermined system of linear equations for the sought-after coefficient $\mathbf{a} \in \mathbb{R}^{(k+c-1)}$ is built. It was proven [11] that this is equivalent to the usage of the pseudoinverse $\mathbf{U}^{(n)\bullet\dagger} = (\mathbf{U}^{(n)^\top}\mathbf{U}^{(n)})^{-1}\mathbf{U}^{(n)^\top} \in \mathbb{R}^{(k+c-1)\times m}$. Using the obtained representation the selected pixels are reconstructed $\mathbf{y}^{(n+1)\bullet} \in \mathbb{R}^p$. Next, pixels with large error, based on the error distribution of the chosen pixels, are removed from the sampled subset. This step is repeated, until a predefined number of reliable pixels is reached (Algorithm 1, steps 3 – 7). Based on these final selected inliers the coefficient is determined (Algorithm 1, step 8).

Since such a pixel set is randomly selected, it may happen that the major amount of chosen points are noisy pixels. In this case the described procedure cannot give a correct result. In order to avoid this problem several hypotheses are built. Consequently, the probability of selecting an adequate subset of pixels increases with the number of calculated hypotheses. The choice of the best hypothesis is based on the corresponding reconstruction errors. All values exceeding a predefined threshold θ (the maximally allowed reconstruction error) are counted. The hypothesis with the smallest error count e, thus, the fewest insufficiently reconstructed pixels, is selected (Algorithm 1, steps 10 – 14).

Algorithm 1 : Robust *ILDAaPCA*

Input: Current robust augmented principal subspace: aPCA vectors $\mathbf{U}^{(n)}$, aPCA coefficients $\mathbf{A}^{(n)}$ and mean vector $\boldsymbol{\mu}^{(n)}$; new input image $\mathbf{x}^{(n+1)}$;

Output: New robust subspaces of pre-specified size: aPCA vectors $\mathbf{U}^{(n+1)}$, aPCA coefficients $\mathbf{A}^{(n+1)}$ and mean vector $\boldsymbol{\mu}^{(n+1)}$; LDA vectors $\mathbf{V}^{(n+1)}$ and class centers $\boldsymbol{\nu}_i^{(n+1)}$

repeat
  Randomly choose a subset of pixels $\mathbf{x}^{\bullet}$ from $\mathbf{x}^{(n+1)} - \boldsymbol{\mu}^{(n)}$.
  repeat
    Calculate the aPCA coefficient from the current subset:
      $\mathbf{a} = \mathbf{U}^{(n)\bullet\dagger}\mathbf{x}^{\bullet}$
    Reconstruct:
      $\mathbf{y}^{\bullet} = \mathbf{U}^{(n)\bullet}\mathbf{a}$
    Remove those pixels with the largest reconstruction error.
  until $\mathbf{x}^{\bullet}$ has a predefined number of pixels.
  Recalculate the aPCA coefficient:
    $\mathbf{a} = \mathbf{U}^{(n)\bullet\dagger}\mathbf{x}^{\bullet}$
until the number of hypothesis is reached.
for each hypothesis **do**
  Calculate the reconstruction error:
    $\varepsilon_i^2 = (x_i - y_i)^2\,,\ \ \forall i \quad \text{where } \mathbf{y} = \mathbf{U}^{(n)}\mathbf{a}$
  Count the values exceeding a predefined threshold θ:
    $e = |\sum_{i=1}^{m} \varepsilon_i^2 > \theta|$
end for
Select the best hypothesis for $\mathbf{a}$ according to the smallest error count e.
repeat
  Reconstruct:
    $\mathbf{y} = \mathbf{U}^{(n)}\mathbf{a}$
  Remove those pixels with the largest reconstruction error.
  Recalculate the aPCA coefficient:
    $\mathbf{a} = \mathbf{U}^{(n)\dagger}\mathbf{x}$
until the desired accuracy (reconstruction error) is reached.
Reconstruct:
  $\mathbf{y}^{(n+1)} = \mathbf{U}^{(n)}\mathbf{a} + \boldsymbol{\mu}^{(n)}$
Calculate the reconstruction error:
  $\varepsilon_i^2 = \left(x_i^{(n+1)} - y_i^{(n+1)}\right)^2\,,\ \ \forall i$
Determine outliers for predefined threshold ϑ:
  $\mathcal{I}^{\circ} = \{i \mid \varepsilon_i^2 > \vartheta\}$
Replace missing pixels:
  $x_i^{(n+1)\circ} = y_i^{(n+1)\circ},\ \ \forall i \in \mathcal{I}^{\circ}$
Add $\mathbf{x}^{(n+1)}$ to current aPCA subspace by Algorithm 2 and Algorithm 3.

In an iterative process the selected pixels for the chosen hypothesis are refined by projection and reconstruction, where pixels with high error are discarded again (Algorithm 1, steps 15 – 19). Based on the final subset of pixels the image $\mathbf{y}^{(n+1)} \in \mathbb{R}^m$ is reconstructed (Algorithm 1, step 20). Then, all pixels with squared reconstruction error ε^2, exceeding a predefined threshold ϑ, are considered as real outliers. These points $x_i^{(n+1)\circ}$ are replaced by their reconstruction $y_i^{(n+1)\circ}$, such that the resulting image can be considered as outlier-free (Algorithm 1, steps 21 – 23).

Finally, it is added to the current subspace using the two step approach presented in [15]. In short, first the aPCA subspace is updated (Algorithm 2) and on this expanded representation the LDA subspace is built (Algorithm 3).

This is realized in a way such that the subspace keeps a predefined size and only the representations are growing with the number of added images: $\mathbf{U}^{(n+1)} \in \mathbb{R}^{m\times(k+c-1)}$, $\mathbf{V}^{(n+1)} \in \mathbb{R}^{(k+c-1)\times(c-1)}$ and $\mathbf{A}^{(n+1)} \in \mathbb{R}^{(k+c-1)\times(n+1)}$.

Algorithm 2 : IPCA on Augmented Basis

Input: Current augmented principal subspace: aPCA vectors $\mathbf{U}^{(n)}$, aPCA coefficients $\mathbf{A}^{(n)}$ and mean vector $\boldsymbol{\mu}^{(n)}$; new (outlier free) input image $\mathbf{x}^{(n+1)}$

Output: New augmented principal subspace: aPCA vectors $\mathbf{U}$, aPCA coefficients $\mathbf{A}$ and mean vector $\boldsymbol{\mu}^{(n+1)}$

1: Project $\mathbf{x}^{(n+1)}$ onto current subspace: $\mathbf{a} = \mathbf{U}^{(n)\top}(\mathbf{x}^{(n+1)} - \boldsymbol{\mu}^{(n)})$
2: Reconstruct: $\mathbf{y} = \mathbf{U}^{(n)}\mathbf{a} + \boldsymbol{\mu}^{(n)}$
3: Compute residual vector: $\mathbf{r} = \mathbf{x}^{(n+1)} - \mathbf{y}$
4: Append new basis vector $\mathbf{r}$: $\mathbf{U}' = \begin{bmatrix} \mathbf{U}^{(n)} & \frac{\mathbf{r}}{\|\mathbf{r}\|} \end{bmatrix}$
5: Determine coefficients in new basis: $\mathbf{A}' = \begin{bmatrix} \mathbf{A}^{(n)} & \mathbf{a} \\ \mathbf{0} & \|\mathbf{r}\| \end{bmatrix}$
6: Perform PCA on $\mathbf{A}'$ obtaining mean value $\boldsymbol{\mu}''$ and eigenvectors $\mathbf{U}''$
7: Project coefficient vectors to new basis: $\mathbf{A} = \mathbf{U}''^{\top}(\mathbf{A}' - \boldsymbol{\mu}''\mathbf{1}_{1\times(n+1)})$
8: Rotate subspace $\mathbf{U}'$ for $\mathbf{U}''$: $\mathbf{U} = \mathbf{U}'\mathbf{U}''$
9: Update mean: $\boldsymbol{\mu}^{(n+1)} = \boldsymbol{\mu}^{(n)} + \mathbf{U}'\boldsymbol{\mu}''$

Algorithm 3 : LDA on Updated Augmented Basis

Input: Updated augmented principal subspace: aPCA vectors $\mathbf{U}$, aPCA coefficients $\mathbf{A}$

Output: New subspaces of pre-specified size: aPCA vectors $\mathbf{U}^{(n+1)}$, aPCA coefficients $\mathbf{A}^{(n+1)}$; LDA vectors $\mathbf{V}^{(n+1)}$ and class centers $\boldsymbol{\nu}_i^{(n+1)}$

1: Perform LDA on $\mathbf{A}$ obtaining LDA vectors $\mathbf{V}$ and class centers $\boldsymbol{\nu}_i^{(n+1)}$
2: Split $\mathbf{V} = \begin{bmatrix} \mathbf{V}_k \\ \mathbf{V}_c \end{bmatrix}$
3: Orthogonalize $\mathbf{V}_c$: $\widetilde{\mathbf{V}}_c = \mathbf{V}_c(\mathbf{V}_c^T\mathbf{V}_c)^{-1/2}$
4: Build projection matrix $\mathbf{M}$: $\mathbf{M} = \begin{bmatrix} \mathbf{I}_k & \mathbf{0}_k \\ \mathbf{0}_c & \widetilde{\mathbf{V}}_c \end{bmatrix}$
5: Project $\mathbf{U}$, $\mathbf{A}$ and $\mathbf{V}$: $\mathbf{U}^{(n+1)} = \mathbf{U}\mathbf{M}$, $\mathbf{A}^{(n+1)} = \mathbf{M}^{\top}\mathbf{A}$, $\mathbf{V}^{(n+1)} = \mathbf{M}^{\top}\mathbf{V}$

3 Experiments

In this section we experimentally confirm that our approach is able to handle the addition of occluded data in an appropriate way. That means, that we can deal with the problem of complete autonomous incremental updating given noisy training data. The task is to detect erroneous pixels as deviations from the data representation following Algorithm 1.

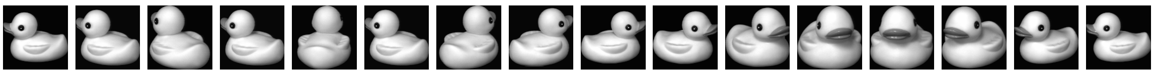
3.1 Data Setup

Two different datasets covering faces as well as other objects were used to analyze the proposed method. Despite the fact, that both datasets consist of the same number of classes the image content differ clearly thus, the final results vary. However, a considerable amount of noise can be handled for both configurations. It will be seen that (as claimed in the theoretical part) the augmented basis really is suitable for a reliable outlier detection always providing satisfactory results.

1. The *pre-cropped Sheffield Face Database* [4] (denoted as SFD) consists of 20 persons with at least 19 images of each individual covering poses from profile to frontal views (e.g., see Figure 2(a)). To have identical class sizes we used exactly 19 images per individual.

2. The *Columbia Image Database Library* [9] (COIL-20) consists of 20 objects with 72 images of views from 0 to 360 degrees in 5 degree steps (e.g., see Figure 2(b)).

(a)



(b)

Figure 2: Sample images of one class from (a) SFD and (b) COIL-100.

The datasets were split into half to get independent training and test data. Actually, the described classes are all structured such that the appearance changes smoothly between the images. In order to have a consistent subspace description we use every second image for training and the remaining ones for testing. While the test data consists of clean images, the training images are corrupted by occlusions. Only those training images used for the starting basis are noise-free, giving a reliable subspace to start from (see Section 2). This reliable subspace builds the fundament for the robust addition of further images. Since our main focus lies on the robustness in the updating step, this assumption is reasonable. Actually, we took two images of each class for the *SFD* dataset and four images for the more complex *COIL-20* dataset, having 40 images and 80 images respectively in total. It is clear that the larger the reliable starting basis the better the representation at the beginning is, leading to a superior performance of the outlier detection. The chosen selection is just for proof of concept. The size k for the reconstructive part of the *aPCA* basis was chosen to encompass 80% of the energy of the given dataset. The energy is defined as a fraction of the total variance (for more details see e.g., [11]). This parameter stays unchanged during the entire experiments. Subsequently, in each iteration one noisy training image (in sequential order) is attached to all classes. The quality of all constructed subspaces is measured by the obtained recognition rate. All presented plots are double tracked in order to allow for a direct comparison of the two databases. Left columns correspond to the results of *SFD*, while right columns correspond to the results of *COIL-20*.

To study the behavior of LDA learning in the presence of occlusions we evaluated several situations. On the one hand, we consider three intensities for the occlusions having black, white and random gray values instead the true pixel values. On the other hand, four types of occlusion are handled, namely salt-and-pepper noise, squares, horizontal bars and vertical bars. All occlusions are added at random positions. Except the salt-and-pepper noise, made up of black and white pixels of equal number, the types of occlusion and the occlusion intensities are mixed such that different possible situations are covered. Example images of different occlusions for both datasets are given in Figure 3 for illustration. The shown amount of added occlusions grows from 0% (noise-free) to 50% in steps of 10%. This percentages refer to the number of image pixels. For better clarity we focus on the case of 20% occlusion in our experiments but also a larger amount could be handled (again see [14]). To further simplify the discussion, we assume that each image contains the same type of occlusion. It will be seen that due to the diverse kinds of images in the databases the utilized occlusions influence the learning process in a different way.

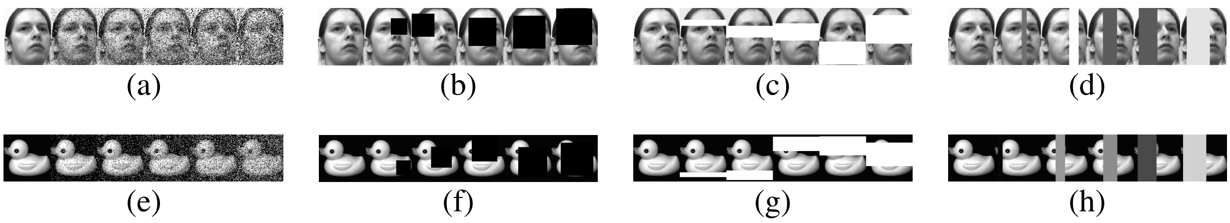


Figure 3: Noisy training data of *SFD* (top) and *COIL-20* (bottom) for 0% to 50% of randomly positioned occlusions: (a,e) Salt-and-pepper, (b,f) Black square, (c,g) White horizontal bar, (d,h) Gray vertical bar.

The main reason why we do not handle background clutter and focus on occlusions only is the amount of noisy image pixels. The performance of our approach (despite being better than the non-robust method) decreases with a growing amount of noise, especially since we rely on a quite small starting basis and analyze the case that all images used in the update steps are corrupted. Consequently, the usage of databases where inconsistent background encompasses large parts of the image will perform rather poor. Furthermore, in order to sample reliable points, the number of hypotheses has to be huge, resulting in a large computational effort. For that matter our method is not optimally suitable to handle background clutter. In this case a different approach, to properly extract the objects from the background has to be applied as a preprocessing step.

3.2 Robust Incremental LDA Results

Figure 4 and Figure 5 present the results for the described setup (having 100 hypotheses, $\theta = 50$ and $\vartheta = 80$ in all experiments). In the plots we compare the achieved robust performance (where missing pixels are approximated by the current representation before they are added to the subspace) to three other approaches:

1. *orig Batch:* builds a new model from scratch in each update step using the same number of images as the incremental algorithm. Since for batch learning all training images are given the full information is available. Thus, it finds the most suitable hyperplanes always yielding the best recognition rates. Here only noise-free images are used in order to get the best recognition rate for the individual datasets achievable using LDA.

2. *known MPL aPCA:* depicts the results that can be obtained for the case that the unreliable pixels are known. These plots give the best possible results that can be achieved with the information represented in the reliable pixels (missing pixels are approximated using reliable pixels only) giving an upper bound for the complete stand-alone method. Note, that if the occlusions are known, the intensities have no effect.

3. *nonrob aPCA:* presents the performance of the non-robust approach (not considering the outliers). It can be observed, that the performances are really poor, emphasizing the need for a proper noise handling. Furthermore it can be seen that the color of the occlusion has a significant influence on the outcome.

For all four described setups (different types of occlusions and different intensities) the robust method approaches the known missing pixel results. In addition, to highlight the benefit of our approach once more the non-robust results are clearly outperformed for both datasets.

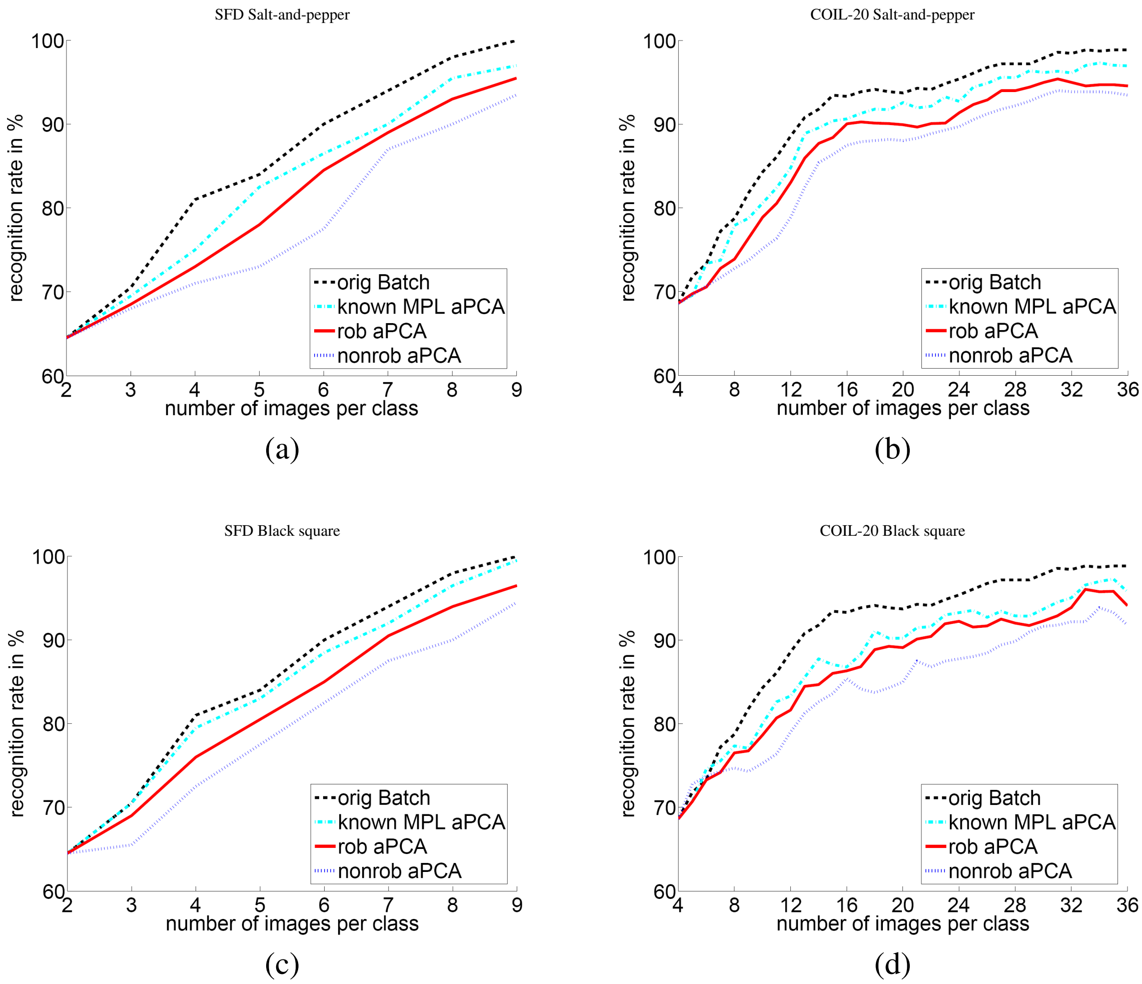
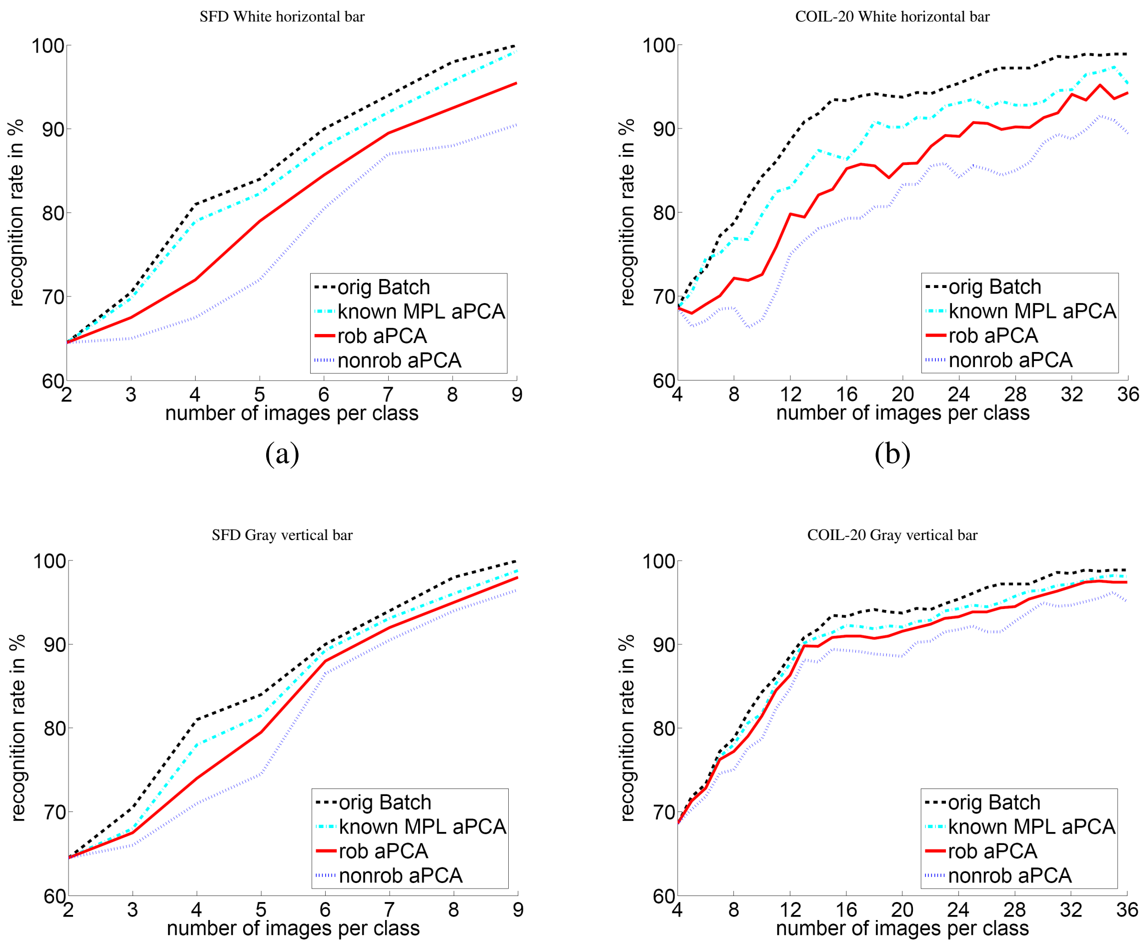


Figure 4: Robust incremental recognition rates for *SFD* (left) and *COIL-20* (right) for 20% of different types of occlusion. Comparison of the results achieved by the noise-free batch learning (orig Batch) to the robust incremental method under the assumption that the missing pixels are known (known MPL aPCA), the complete robust incremental method (rob aPCA) and the non-robust incremental method (nonrob aPCA): (a,b) Salt-and-pepper noise and (c,d) Black square.



(a) (b) (c) (d)

Figure 5: Robust incremental recognition rates for *SFD* (left) and *COIL-20* (right) for 20% of different types of occlusion. Comparison of the results achieved by the noise-free batch learning (orig Batch) to the robust incremental method under the assumption that the missing pixels are known (known MPL aPCA), the complete robust incremental method (rob aPCA) and the non-robust incremental method (nonrob aPCA): (a,b) White horizontal bar and (c,d) Gray vertical bar.

Finally, Figure 6 shows some examples of detected outliers and their reconstruction for the SFD database. Actually, these are the first images to be added to the basis built from noise free data. Inaccuracies in the reconstruction result from the small number of training data describing the current subspace. Nevertheless, the eventually added image is of much better quality than the noisy one.

The selected outliers in Figure 6(b,f,i,l) are also the reason for the differences between the performance of the robust approach and the results of the known missing pixel case. It can be seen that on the one hand not all outliers are detected while on the other hand several reliable pixels are considered as being noise (this is due to the fact that the views are changing which is not properly represented in the current subspace). In case of the known missing pixels, this influence is less significant (the reconstruction is not perfect either but no additional error evolves).

In order to further increase the results of the robust approach there are two essential possibilities. As already mentioned, in all the experiments identical values were used for the thresholds in Algorithm 1. This was just for simplicity and for showing the possibilities of our approach. Thus, a fine-tuning of these parameters will improve the results. In addition the bases we start from are quite small not representing the variation in the data very well. If there are more training images at the beginning that are better spread over the different views the robust approach would profit clearly (finding more positive and less negative outliers) while the known missing pixel case would improve only minimal from the better reconstruction.

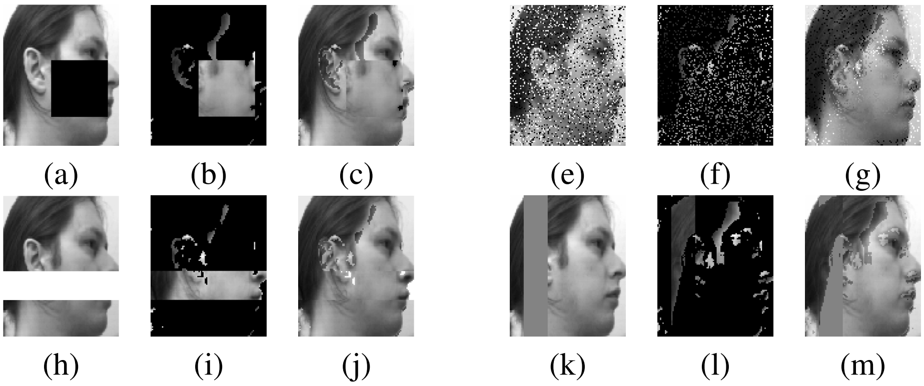


Figure 6: Examples for detected outliers for the SFD dataset: (a,e,h,k) Noisy images, (b,f,i,l) Detected outliers and (c,g,j,m) Outliers replaced by their reconstruction.

4 Conclusion

The analysis of the influence of different types of occlusion and intensities for different datasets showed that bringing robustness into the incremental LDA training is a very important issue. Strictly speaking, it was shown how the updating is negatively influenced by incorrect data. We proposed a method that automatically finds outlying pixels in order to ensure a correct extension of the subspace. The additional reconstructive information incorporated in our LDA representation enables the detection of these outliers by analyzing the consistency of the training data. Plain discriminative methods cannot use this type of information, while plain reconstructive approaches discard important classification information. The experiments proved that the properties of the *aPCA* basis allow for a correct approximation of the missing information. Moreover, as long as the amount of noise is reasonable we yield similar results as if the training was performed on clean data. As a summary, it can be said that our approach autonomously can handle pixel noise in images used for updating the subspace, always providing a convincing performance.

References

[1] H. Bischof and A. Leonardis. Recognizing Objects by Their Appearance Using Eigenimages. In V. Hlavac, K. G. Jeffery, and J. Wiedermann, editors, *SOFSEM 2000: Theory and Practice of Informatics: 27th Conference on Current Trends in Theory and Practice of Informatics*, volume 1963 of *Lecture Notes in Computer Science*, pages 245 – 265. Springer, November – December 2000.

[2] S. Fidler, D. Skočaj, and A. Leonardis. Combining Reconstructive and Discriminative Subspace Methods for Robust Classification and Regression by Subsampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):337 – 350, March 2006.

[3] R. A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7:179–188, 1936.

[4] D. B. Graham and N. M. Allinson. *Face recognition: From theory to applications*, volume 163 of *NATO ASI Series F, Computer and Systems Sciences*, chapter Characterizing virtual eigensignatures for general purpose face recognition, pages 446 – 456. Springer, 1998.

[5] H. Hotelling. Analysis of a Complex of Statistical Variables into Principal Components. *Journal of Educational Psychology*, 24:417 – 441, 1933.

[6] S. Kim, A. Magnani, and S. P. Boyd. Robust Fisher Discriminant Analysis. In Y. Weiss and B. Schölkopf and J. Platt, editor, *Advances of the 18th Conference on Neural Information Processing Systems (NIPS 2005)*, volume 18, pages 659 – 666, December 2005.

[7] A. Leonardis and H. Bischof. Robust Recognition Using Eigenimages. *Computer Vision and Image Understanding*, 78(1):99 – 118, 2000.

[8] Y. Li, J. Jiang, Z. Chen, C. Xu, and R. Yu. Robust Linear Discriminant Analysis for Chemical Pattern Recognition. *Journal of Chemometrics*, 13(1):3 – 13, April 1999.

[9] S. A. Nene, S. K. Nayar, and H. Murase. Columbia Object Image Library (COIL-20). Technical Report CUCS-005-96, Columbia University, February 1996.

[10] V. Roth. Kernel Fisher Discriminants for Outlier Detection. *Neural Computation*, 18(4):942 – 960, April 2006.

[11] D. Skočaj. *Robust Subspace Approaches to Visual Learning and Recognition.* PhD thesis, University of Ljubljana, Faculty of Computer and Information Science, 2003.

[12] D. Skočaj and A. Leonardis. Weighted and Robust Incremental Method for Subspace Learning. In *Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV 2003)*, volume 2, pages 1494 – 1501, October 2003.

[13] V. Todorov and A. M. Pires. Comparative Performance of Several Robust Linear Discriminant Analysis Methods. *REVSTAT - Statistical Journal*, 5(1):63 – 83, March 2007.

[14] M. Uray. *Incremental, Robust, and Efficient Linear Discriminant Analysis Learning.* PhD thesis, Graz University of Technology, Institute for Computer Graphics and Vision, 2008.

[15] M. Uray, D. Skočaj, P. M. Roth, H. Bischof, and A. Leonardis. Incremental LDA Learning by Combining Reconstructive and Discriminative Approaches. In Abhir Bhalerao and Nasir Rajpoot, editor, *Proceedings of the 18th British Machine Vision Conference (BMVC 2007)*, volume 1, pages 272 – 281, September 2007.