EFFICIENT CLASSIFICATION FOR LARGE-SCALE PROBLEMS BY MULTIPLE LDA SUBSPACES

Martina Uray
Institute of Digital Image Processing
Joanneum Research
Martina.Uray@joanneum.at

Peter M. Roth, Horst Bischof
Institute for Computer Graphics and Vision
Graz University of Technology
{*pmroth, bischof*}*@icg.tugraz.at*

Keywords: LDA, image classification, large databases

Abstract: In this paper we consider the limitations of Linear Discriminative Analysis (LDA) when applying it for large-scale problems. Since LDA was originally developed for two-class problems the obtained transformation is sub-optimal if multiple classes are considered. In fact, the separability between the classes is reduced, which decreases the classification power. To overcome this problem several approaches including weighting strategies and mixture models were proposed. But these approaches are complex and computational expensive. Moreover, they were only tested for a small number of classes. In contrast, our approach allows to handle a huge number of classes showing excellent classification performance at low computational costs. The main idea is to split the original data into multiple sub-sets and to compute a single LDA space for each sub-set. Thus, the separability in the obtained subspaces is increased and the overall classification power is improved. Moreover, since smaller matrices have to be handled the computational complexity is reduced for both, training and classification. These benefits are demonstrated on different publicly available datasets. In particular, we consider the task of object recognition, where we can handle up to 1000 classes.

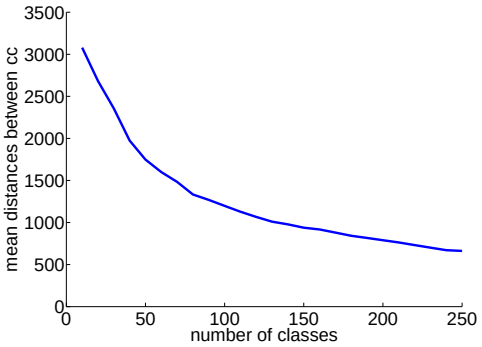
1 INTRODUCTION

Linear Discriminative Analysis (LDA) is a popular and widely used statistical technique for dimension reduction and linear classification. Important applications include face recognition (Belhumeur et al., 1997), speech recognition (Hunt, 1979), or image retrieval (Swets and Weng, 1996). The main idea is to search for a linear projection, that preserves a maximum amount of discriminative information when projecting the original data onto a lower dimensional space. In fact, Fisher (Fisher, 1936) introduced a projection that minimizes the Bayes error for two classes. Hence, by maximizing the Fisher criterion (see Section 2), that analyzes the between scatter versus the within scatter for two classes, an optimal solution with respect to the Bayes error is obtained. For more details see, e.g., (Fukunaga, 1990).

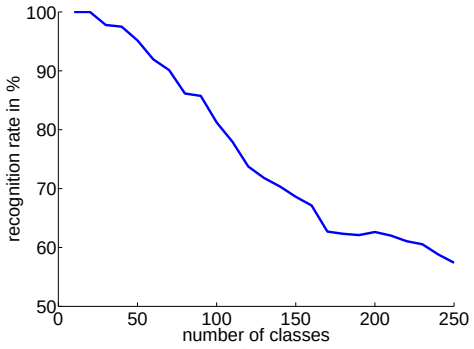
Later this approach was extended for multiple classes by Rao (Rao, 1948). But Loog et al. (Loog et al., 2001) showed that for more than two classes maximizing the Fisher criterion provides only a sub-optimal solution. In general, the Fisher criterion maximizes the mean squared distances between the classes, which, however, is different from minimizing the Bayes error. In particular, thus obtained projections tend to overemphasize distances of already well separable classes (in the original space). Neighboring classes may overlap in the projected subspace, which reduces the separability and the classification performance. But for many practical applications (e.g., face recognition) only a small number of well separable classes are considered. Under this condition even the sub-optimal solution mostly provides an approximation of sufficient accuracy to solve the specific task.

In contrast, in this paper, we apply LDA for multi-class classification for large-scale problems (i.e., up to 1000 classes). Hence, we expect that due to the sub-optimal projection an increasing number of classes would decrease the classification performance. This is illustrated for an image classification task (i.e., on the *ALOI* database (Geusebroek et al., 2005)). From Figure 1(a) it can be seen that for an increasing number of classes (i.e, starting from 10 up to 250) the dis-

tances between the class centers and thus the separability are decreased. As a consequence, the classification rate is successively decreased, which is shown in Figure 1(b).



(a)



(b)

Figure 1: Decreasing classification performance for increasing number of classes: (a) mean distances between class centers and (b) related corresponding classification rates.

To overcome these drawbacks several approaches were proposed that are based on implicit weighting schemes (Zhou and Yang, 2004; Loog et al., 2001) or the estimation of mixture models (Kim et al., 2003). Loog et al. (Loog et al., 2001) introduced a modified Fisher criterion that is more closely related to the classification error. For that purpose, they decompose the c-class criterion into $\frac{1}{2}c(c-1)$ two-class criteria. This allows to define weighting functions penalizing classes that are close together, such that the contribution of each pair of classes for the overall criterion directly depends on the Bayes error between the two classes (*weighted pairwise Fisher criteria*). But due to its computational complexity and the occurrence of the small sample size problem this approach can not directly be applied for high-dimensional data such as images. Thus, Zhou and Yang (Zhou and Yang, 2004) reduce the dimension of the input data by discarding the null-space of the between-class scatter matrix and apply the weighted LDA approach on the thus dimension reduced data. In contrast, Kim et al. (Kim et al., 2002; Kim et al., 2003) propose to estimate LDA mixture models (especially, to cope with multi-modal distributions). The main idea is to apply PCA mixture models (Tipping and Bishop, 1999) to cluster individual classes first. Then, individual LDA projection matrices are estimated and classification is done by the standard nearest neighbor search over all projections.

The methods described above reduce the problems resulting from the sub-optimal projection, but they have two disadvantages. First, they are computational very expensive. For the weighted LDA approaches the pairwise Fisher criteria have to be estimated for all pairs. Similarly, for the LDA mixture approach the PCA mixture models have to be estimated in an iterative way using the EM-algorithm (Dempster et al., 1977). In both cases these computations might be quite expensive, especially, if the number of classes is very large. Second, these methods were only evaluated for small datasets (i.e., up to 128 classes). Hence, in this paper we propose a method that is computationally much cheaper; even for very large-scale problems!

The main idea is to reduce the complexity of the problem by splitting the data into a pre-defined number of equal sized sub-clusters that can still be handled by a single LDA model. Since, in this work we are mainly focused on reducing the problem's complexity these clusters are selected randomly. Once the LDA subspaces were estimated an unknown sample can be classified by projecting it onto all subspaces. The classification is finally done by a nearest neighbor search. Since the subspaces are isometric the Euclidean norm is equivalent over all subspaces and the closest class center can be chosen. As shown in the experiments, in this way the classification performance can significantly be improved; especially if the number of classes is very large. Moreover, since smaller matrices have to be handled the computational costs as well as the memory requirements can be dramatically reduced; especially in the training stage!

The outline of the remaining paper is as follows: First, in Section 2 we review the standard LDA approach and introduce the multiple LDA subspace representation. Next, experimental results are given in Section 3. Finally, we conclude the paper in Section 4.

2 MULTIPLE LDA SUBSPACES

2.1 Standard LDA

Given a dataset $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$ of n samples, where each sample belongs to one of c classes $C_1, \ldots, C_c$. Then, LDA computes a classification function

$$g(\mathbf{x}) = \mathbf{W}^\top \mathbf{x}\,, \tag{1}$$

where $\mathbf{W}$ is selected as the linear projection, that minimizes the within-class scatter

$$\mathbf{S}_w \in \mathbb{R}^{m \times m} = \sum_{i=1}^{c} n_i (\mu_i - \mu)(\mu_i - \mu)^\top \tag{2}$$

whereas it maximizes the between-class scatter

$$\mathbf{S}_b \in \mathbb{R}^{m \times m} = \sum_{i=1}^{c} \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mu_i)(\mathbf{x} - \mu_i)^\top\,, \tag{3}$$

where μ is the mean over all samples, μ_i is the mean over class C_i, and n_i is the number of samples in class C_i. In fact, this projection is obtained by maximizing the Fisher-criterion

$$\mathbf{W}_{opt} = \arg\max_{\mathbf{W}} \frac{\left|\mathbf{W}^\top \mathbf{S}_b \mathbf{W}\right|}{\left|\mathbf{W}^\top \mathbf{S}_w \mathbf{W}\right|}\,. \tag{4}$$

The optimal solution for this optimization problem is given by the solution of the generalized eigenproblem

$$\mathbf{S}_b \mathbf{W} = \Lambda \mathbf{S}_w \mathbf{W}\,, \tag{5}$$

or directly by computing the eigenvectors for $\mathbf{S}_w^{-1}\mathbf{S}_b$. Since the rank of $\mathbf{S}_w^{-1}\mathbf{S}_b$ is bounded by the rank of $\mathbf{S}_b$ there are $c-1$ non-zero eigenvalues resulting in a $(c-1)$-dimensional subspace $\mathbf{L} = \mathbf{W}^\top \mathbf{X} \in \mathbb{R}^{(c-1) \times n}$, which preserves the most discriminant information. For classification of a new sample $\mathbf{x} \in \mathbb{R}^m$ the class label $\omega \in \{1, \ldots, c\}$ is assigned according to the result of a nearest neighbor classification. For that purpose, the Euclidean distances d of the projected sample $g(\mathbf{x})$ and the class centers $\nu_i = \mathbf{W}^\top \mu_i$ in the LDA space are compared:

$$\omega = \arg\min_{1 \le i \le c} d\left(g(\mathbf{x}), \nu_i\right)\,. \tag{6}$$

2.2 Multiple Class LDA

To overcome the limitations of the standard LDA approach when applying it for large-scale problems we propose to reduce the complexity by splitting the problem into sub-problems. This can be motivated on basis of a theoretic criterion proposed by Martínez and Zhu (Martínez and Zhu, 2005), that describes the linear separability of classes:

$$\widetilde{K} \;=\; \frac{1}{c-1} \sum_{i=1}^{c-1} \max_{\forall r \le s} \left(\mathbf{u}_r^\top \mathbf{v}_s\right)^2\,. \tag{7}$$

The parameter $\widetilde{K}$ is estimated by analyzing the angle between the eigenvectors $\mathbf{u}_r$ and $\mathbf{v}_s$, obtained by solving the eigenproblem for the scatter matrices $\mathbf{S}_b$ and $\mathbf{S}_w$:

$$\mathbf{S}_b \mathbf{U} \;=\; \Lambda_u \mathbf{U} \tag{8}$$

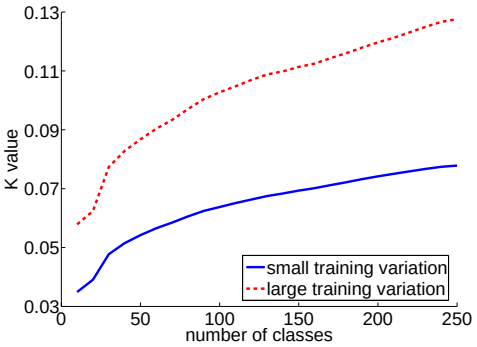
$$\mathbf{S}_w \mathbf{V} \;=\; \Lambda_v \mathbf{V}\,. \tag{9}$$

In fact, it was shown that a large value $\widetilde{K}$, which is equivalent to a small angle, corresponds to a high probability of incorrect classification. To investigate the separability for an increasing number of classes we analyzed the parameter $\widetilde{K}$ for varying datasets of different complexity. In particular, we considered growing sub-sets of *ALOI* (in steps of ten classes) for different levels of variability in the training data (i.e., small versus large changes between the training views). From Figure 2(a) it can be seen that increasing the complexity of the problem (i.e., increasing the number of classes) also increases $\widetilde{K}$. That is, the separability and thus, as illustrated in Figure 2(b), the classification performance is decreased. Moreover, it can be seen that in addition to the number of classes the similarity between training and testing data has also a large influence on the separability.

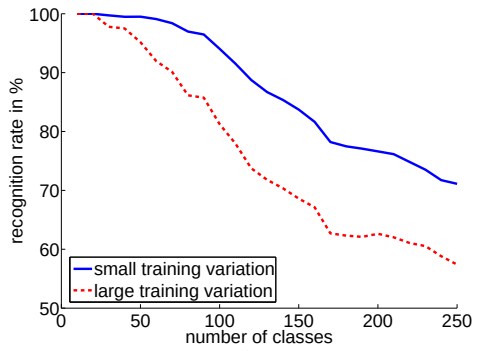
From this observation we can deduce that simplifying the complexity of the problem by reducing the number of classes increases the classification power. Consequently, instead of building a single large subspace we propose to split the data into several subclusters and to build multiple small (well separable) subspaces.

More formally, given a dataset $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$ of n samples, where each sample belongs to one of c classes $C_1, \ldots, C_c$. First, $\mathbf{X}$ is randomly split into k non-overlapping clusters $\mathbf{X}_j \in \mathbb{R}^{m \times n_j}$, each consisting of l classes $C_{j1}, \ldots, C_{jl}$, such that $\mathbf{X} = \{\mathbf{X}_1 \cup \ldots \cup \mathbf{X}_k\}$. In addition, the class means $\{\mu_{j1}, \ldots, \mu_{jl}\} \subset \{\mu_1, \ldots, \mu_c\}$ and the overall cluster mean μ_j are estimated. Next, for each cluster $\mathbf{X}_j$ an $(l-1)$-dimensional LDA subspace $\mathbf{L}_j = \mathbf{W}_j^\top \mathbf{X}_j \in \mathbb{R}^{(l-1) \times n_j}$ is estimated by solving the individual eigenvalue problems

$$(\mathbf{S}_{wj}^{-1}\mathbf{S}_{bj})\mathbf{W}_j = \Lambda_j \mathbf{W}_j\,, \tag{10}$$



(a)



(b)

Figure 2: Decreasing linear separability for increasing number of classes for varying complexity of the training data: (a) Quality criterion $\widetilde{K}$ and (b) corresponding recognition rates.

where

$$\mathbf{S}_{wj} = \sum_{i=1}^{l} \sum_{\mathbf{x} \in C_{ji}} (\mathbf{x}_j - \mu_{ji})(\mathbf{x}_j - \mu_{ji})^\top \quad (11)$$

$$\mathbf{S}_{bj} = \sum_{i=1}^{l} n_{ji} (\mu_{ji} - \mu_j)(\mu_{ji} - \mu_j)^\top . \quad (12)$$

Finally, for each LDA subspace l class centers $\nu_{ji} \in \mathbb{R}^{l-1}$ are determined by projecting class means μ_{ji} onto the corresponding subspaces:

$$\nu_{ji} = \mathbf{W}_j^\top \mu_{ji}, \;\; i = 1, \ldots, l . \quad (13)$$

Each of the k subspaces internally describes l classes C_{ji}. Thus, the class centers ν_{ji} have to be reassigned to their original labels. Since there is no intersection of clusters (i.e., each class is assigned to exactly one cluster) the relabeling is unique, giving the original number of c class centers:

$$\bigcup_{j=1}^{k} \{\nu_{j1}, \ldots, \nu_{jl}\} \mapsto \{\nu_1, \ldots, \nu_c\} . \quad (14)$$

The whole training procedure is summarized in Algorithm 1.

Algorithm 1 : Multiple LDA learning

Input: Dataset $\mathbf{X} \in \mathbb{R}^{m \times n}$, number of sub-clusters k and data labels $\omega \in \{1, \ldots, c\}$

Output: LDA matrices $\mathbf{W}_j \in \mathbb{R}^{m \times (l-1)}$ and class centers $\{\nu_1, \ldots, \nu_c\}$ with $\nu_i \in \mathbb{R}^{(l-1)}$

1: Sample $l = c/k$ random objects for each cluster:
$\{\mathfrak{L}_j\} \subset \{1, \ldots, c\}, \;\; |\mathfrak{L}_j| = l$
and $\{\mathfrak{L}_r\} \cap \{\mathfrak{L}_s\} = \{\}, \;\; \forall r \neq s$

2: Split the dataset:
$\mathbf{X}_j \subset \mathbf{X}$ according to $\{\mathfrak{L}_j\}, \;\; \mathbf{X}_j \in \mathbb{R}^{m \times n_j}$

3: **for** $j = 1$ to k **do**

4: Calculate PCA+LDA on each dataset $\mathbf{X}_j$:
$\mathbf{W}_j \in \mathbb{R}^{m \times (l-1)}$

5: Project class means onto LDA space:
$\nu_{ji} = \mathbf{W}_j^\top \mu_{ji}, \;\; i = 1, \ldots, l$

6: **end for**

7: Reassign sub-cluster class centers the overall class label:
$\bigcup_{j=1}^{k} \{\nu_{j1}, \ldots, \nu_{jl}\} \mapsto \{\nu_1, \ldots, \nu_c\}$

Once the k LDA subspaces were estimated the crucial step is the classification. For that purpose a test sample $\mathbf{x} \in \mathbb{R}^m$ is projected onto all k subspaces. Since all clusters are of the same size the resulting (Euclidean) LDA subspaces are of the same dimension. From Linear Algebra it is known that vector spaces of the same dimension are isomorphic[1] and that isomorphic structures are structurally identical (see, e.g., (Strang, 2006)). Hence, the Euclidean distances in these subspaces accord and can directly be compared. Thus, the class label $\omega \in \{1, \ldots, c\}$ for an unknown test sample $\mathbf{x}$ can be estimated by searching for the closest class center over all projected spaces:

$$g_j(\mathbf{x}) = \mathbf{W}_j^\top \mathbf{x} \quad (15)$$

$$\omega = \arg\min_{1 \leq i \leq c, 1 \leq j \leq k} d(g_j(\mathbf{x}), \nu_i) . \quad (16)$$

[1] An isomorphism is a one-to-one map from a vector space onto itself.

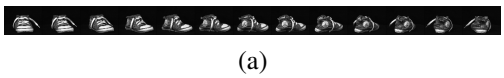
3 EXPERIMENTAL RESULTS

In this section, we show the benefits of the proposed approach. First, we discuss the typical problems that occur when the number of classes is increasing. For that purpose, we analyze the distances between the class centers in relation to the number of classes and discuss the influence of how the subclusters are selected. Next, we compare the proposed approach with a standard LDA approach, which encompasses only a single subspace[2]. In fact, we will show that our method can handle large databases (even with large variability in the appearance) considerably better than the standard approach. Finally, we show that there is also a benefit in terms of memory requirements and computational costs.

The experiments were carried out on two large publicly available databases, that were slightly adapted:

1. The *Columbia Image Database Library* (*COIL-100*) (Nene et al., 1996) consists of 100 objects with 72 colored images of views from 0 to 360 degrees taken in 5 degree steps.
2. The *Amsterdam Library Of Images ALOI-1000* (Geusebroek et al., 2005) consists of 1000 objects with 72 colored images of views from 0 to 360 degrees taken in 5 degree steps. The images contain the objects in their original size as well as some background. To define tasks of different complexity, we created two additional datasets: *ALOI-100*, which consists of the first 100 objects, and *ALOI-250*, which consists of 250 randomly chosen objects.

In order to emphasize the influence of data complexity we used training images describing viewpoint changes of 15° and 30° having two levels of difficulty (see Figure 4). All other images were used for testing, respectively.



(a)

(b)

Figure 3: Training images for one object of *ALOI*: (a) 15° dataset and (b) 30° dataset.

[2]We are using the PCA+LDA approach (Belhumeur et al., 1997) for subspace construction.

3.1 Cluster Analysis

As discussed in Section 2 an increasing number of classes encompassed by a single subspace results in decreasing distances of the class centers. Moreover, the distances between the class centers and the (test) samples are increasing. Both, in fact, reduce the separability between the classes. This behavior is illustrated in Table 1, where the mean distances between the class centers and the mean distances between the test data and the class centers are listed for the 15° and the 30° datasets.

dataset	center to center	test to center
COIL-100	267.78	116.38
ALOI-100	479.71	88.90
ALOI-250	294.21	240.76
ALOI-1000	167.02	89.48

(a)

dataset	center to center	test to center
COIL-100	509.74	251.14
ALOI-100	718.27	188.71
ALOI-250	661.92	477.46
ALOI-1000	375.75	326.67

(b)

Table 1: Mean distances between class centers and between test data and correctly assigned class centers (single subspace LDA): (a) 15° datasets and (b) 30° datasets.

It can be seen that the mean distances between the class centers are reduced to half when the number of classes is decoupled. Similarly, the mean distance between the test data and the class centers is doubled. This shows clearly that the final classifications get increasingly unreliable. Thus, it is clear that smaller clusters would give better results!

In order to build appropriate clusters we analyzed which cluster size would give the best recognition results. For that purpose, we divided each of the datasets randomly into all valid sub-clusters (i.e, all subspaces have to have the same size). For instance, for *ALOI-100* we got $l \in \{2,4,5,10,20,25,50\}$. The obtained recognition rates for all datasets are depicted in Figure 4. It can be seen that the maxima lie between 10 and 25 objects per cluster. Since there are no significant differences between the classification errors in this range the selection of $l = 10$ is reasonable (also other datasets showed good performance using this sub-cluster size). The drop of the recognition rates for growing subspaces emphasizes the observation that a small subspace better spans the cluster centers of the objects, reducing the probability of a fail in classifica-

tion. On the contrary, if the number of sub-clusters is too small the recognition is not reliable either. This results from the fact that the classes mutually influence each other when building the subspace such that an unknown sample has no clear class correspondence. But if there are only two classes one class center is always favored, being affected by only one other class. As a result the closest distances in all clusters are similar and a correct assignment is not possible.

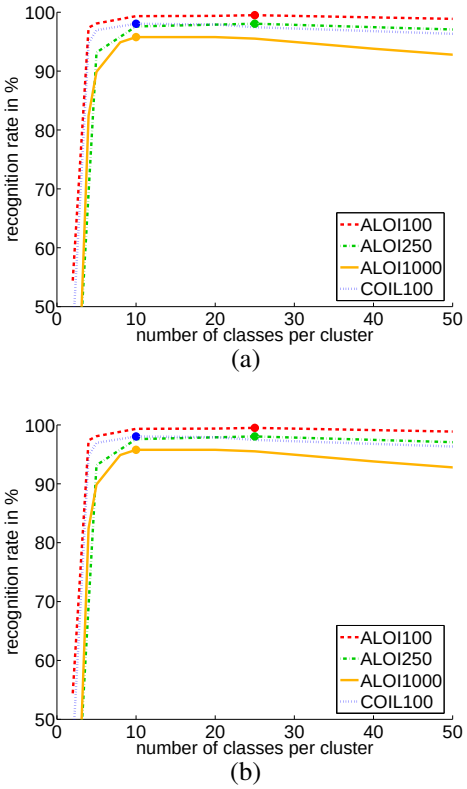


Figure 4: Training images for one object of *ALOI*: (a) 15° dataset and (b) 30° datasets.

In this work we are mainly concerned with reducing the complexity of the classification problem. Thus, the sub-clusters were selected randomly. But in the following we show that the selection of the clusters has only little influence on the performance of our approach. For that purpose, we randomly split the three different datasets of *ALOI* into sub-clusters of size 10 and applied the proposed multiple subspace method. This procedure was repeated ten times; having different clusters each time! The results are summarized in Figure 5.

From the box-plots it can be seen that the variance in the recognition rates for varying selections is quite small (i.e., approx. $\pm 0.5\%$ for the 15° datasets

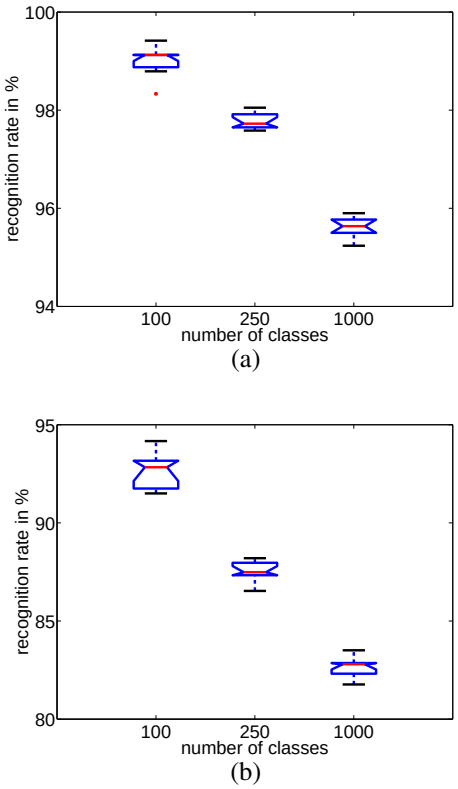


Figure 5: Box plots of the recognition rates of the *ALOI* datasets by repeated random sampling of sub-clusters containing 10 objects each: (a) 15° datasets and (b) 30° datasets.

and approx. $\pm 1.0\%$ for the 30° datasets). Thus, the composition of the clusters has only little influence on the recognition rate and the discrimination between arbitrary classes is mainly sensitive to the number of classes.

3.2 Classification Results

The crucial step in the multiple subspace recognition is that the assignment of clusters implicitly also returns the "hidden" nearest class center. Thus, each sub-cluster label has to be mapped to the overall class label, which, in fact, is a simple lookup. As can be seen from Table 2 the cluster assignment works very well for both training scenarios (i.e., 15° and 30° datasets), resulting in a much higher recognition rate than for the single LDA subspace. More precisely, the cases, where the correct clusters are found but the objects are miss-classified, are quite rare. In addition, in Table 2 we give a comparison of results obtained

using the proposed approach employing sub-clusters of size 10 compared to a standard LDA approach for four datasets of different complexity.

dataset	correct cluster	multiple LDA spaces	single LDA space
COIL-100	98.04 %	98.04%	85.79%
ALOI-100	99.75 %	99.75%	96.25%
ALOI-250	97.58 %	97.52%	71.12%
ALOI-1000	95.77 %	95.76%	62.24%

(a)

dataset	correct cluster	multiple LDA spaces	single LDA space
COIL-100	90.75%	89.75%	69.67%
ALOI-100	93.58%	93.08%	87.42%
ALOI-250	89.73%	88.77%	57.40%
ALOI-1000	84.12%	83.97%	37.31%

(b)

Table 2: Classification performance: correct sub-cluster and final recognition rate (10 objects per cluster) vs. recognition rate in full LDA space: (a) 15° datasets and (b) 30° datasets.

It can be seen that the proposed method outperforms the single subspace LDA for all datasets and that the recognition rate can be drastically increased. Especially, for the different subsets of the *ALOI* database the relative improvement of the final recognition rate is increasing with increasing complexity of the task. In fact, for the *ALOI-1000* we finally achieve an improvement of 34% for the 15° dataset and even 47% for the 30° dataset. Similar results (i.e., a recognition rate of $94\% - 98\%$) for such large image datasets (i.e., *ALOI-1000*) were only reported by Kim et al. (Kim et al., 2007). But the results can not be directly compared, since the database was adapted slightly different (i.e., only 500 classes were considered and the training and test sets were defined differently).

3.3 Memory Requirements and Computation Time

An additional advantage of smaller subspaces is that smaller matrices have to be handled, resulting in lower memory requirements and reduced computational costs. This especially credits for the training, which heavily depends on the eigenvalue decomposition of the training data in the PCA step and of the scatter matrices in the LDA step. This is demonstrated in Table 3 and in Table 4, where we summarized the computation times[3] and memory requirements for the 30° datasets, respectively.

dataset	multiple LDA spaces	single LDA space
COIL-100	27.66s	103.57s
ALOI-100	16.41s	105.51s
ALOI-250	41.22s	1414.24s
ALOI-1000	166.00s	27395.22s

(a)

dataset	multiple LDA spaces	single LDA space
COIL-100	3.85s	7.14s
ALOI-100	6.41s	11.64s
ALOI-250	39.13s	72.88s
ALOI-1000	426.78s	733.80s

(b)

Table 3: Computational costs for the 30° datasets: (a) training and (b) testing.

dataset	multiple LDA space	single LDA spaces
COIL-100	138.54MB	221.68MB
ALOI-100	161.07MB	296.56MB
ALOI-250	254.38MB	655.78MB
ALOI-1000	540.52MB	3331.25MB

(a)

dataset	multiple LDA spaces	single LDA space
COIL-100	227.15MB	257.47MB
ALOI-100	215.35MB	217.90MB
ALOI-250	489.22MB	556.30MB
ALOI-1000	2089.10MB	2175.41MB

(b)

Table 4: Memory requirements for the 30° datasets: (a) training and (b) testing.

It can be seen that for increasing complexity the relative computation time for training compared to the standard approach is reduced. Starting from a reduction factor 4 for *COIL-100* to a factor 165(!) for *ALOI-1000*! During evaluation the effect is less significant since the total representation size is only slightly reduced (i.e., from $c-1$ to $c-k$). But still the evaluation effort is approximately halved.

[3]The experiments were carried out in MATLAB on an Intel Xeon 3.00GHz machine with 8GB RAM.

The same applies for the memory requirements – increasing the complexity relatively decreases the costs. In particular, for *ALOI-1000* the required memory for training was reduced from more than 3GB to 540MB by a factor 6. But there are no significant differences for the evaluation stage since due to the implementation all test data is kept in memory.

4 CONCLUSION

In this paper we presented an approach that overcomes the main limitations when applying LDA for a large number of classes. The main idea is to (randomly) split the original data into several subsets and to compute a separate LDA representation for each of them. To classify a new unknown test sample it is projected onto all subspaces, where a nearest neighbor search is applied to assign it to the correct cluster and hidden class, respectively. To demonstrate the benefits of our approach we applied it for two large publicly available datasets (i.e., *COIL-100* and *ALOI*). In fact, compared to a single model LDA we get a much better classification results, which are even competitive for large datasets containing up to 1000(!) classes. Moreover, since the resulting data matrices are much smaller the memory requirements and the computational costs are dramatically reduced. Future work will include to apply a more sophisticated clustering, which, in fact, would further increase the separability and thus the classification power of the method.

REFERENCES

Belhumeur, P. N., Hespanha, J. P., and Kriegman, D. J. (1997). Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):711 – 720.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Royal Statistical Society*, 39(1):1 – 38.

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

Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. Academic Press.

Geusebroek, J. M., Burghouts, G. J., and Smeulders, A. W. M. (2005). The Amsterdam Library of Object Images. *Computer Vision*, 61(1):103 – 112.

Hunt, M. (1979). A statistical approach to metrics for word and syllable recognition. In *Meeting of the Acoustical Society of American*, volume 66, pages 535 – 536.

Kim, H., Kim, D., and Bang, S. Y. (2002). Face Recognition Using LDA Mixture Model. In *Proc. Intern. Conf. on Pattern Recognition*, volume 2, pages 486 – 489.

Kim, H., Kim, D., and Bang, S. Y. (2003). Extensions of LDA by PCA Mixture Model and Class-Wise Features. *Pattern Recognition*, 36(5):1095 – 1105.

Kim, T.-K., Kittler, J., and Cipolla, R. (2007). Discriminative Learning and Recognition of Image Set Classes Using Canonical Correlations. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(6):1005 – 1018.

Loog, M., Duin, R. P. W., and Haeb-Umbach, R. (2001). Multiclass Linear Dimension Reduction by Weighted Pairwise Fisher Criteria. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(7):762 – 766.

Martínez, A. M. and Zhu, M. (2005). Where are Linear Feature Extraction Methods Applicable? *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(12):1934 – 1944.

Nene, S. A., Nayar, S. K., and Murase, H. (1996). Columbia Object Image Library (COIL-100). Technical Report CUCS-006-96, Columbia University.

Rao, C. R. (1948). The Utilization of Multiple Measurements in Problems of Biological Classification. *Royal Statistical Society – Series B*, 10(2):159 – 203.

Strang, G. (2006). *Linear Algebra and Its Applications*. Brooks/Cole.

Swets, D. L. and Weng, J. (1996). Using Discriminant Eigenfeatures for Image Retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(8):831 – 837.

Tipping, M. E. and Bishop, C. M. (1999). Mixtures of Probabilistic Principal Component Analyzers. *Neural Computation*, 11(2):443 – 482.

Zhou, D. and Yang, X. (2004). Face Recognition Using Direct-Weighted LDA. In *Proc. of the Pacific Rim Int. Conference on Artificial Intelligence*, pages 760 – 768.