Visual object categorization using distance-based discriminant analysis

Serhiy Kosinov, Stéphane Marchand-Maillet,Thierry Pun
Computer Vision and Multimedia Lab, University of Geneva
24 rue du General-Dufour, Geneva, Switzerland
{kosinov,marchand,pun}@cui.unige.ch

Abstract

This paper formulates the problem of object categorization in the discriminant analysis framework focusing on transforming visual feature data so as to make it conform to the compactness hypothesis in order to improve categorization accuracy. The sought transformation, in turn, is found as a solution to an optimization problem formulated in terms of inter-observation distances only, using the technique of iterative majorization. The proposed approach is suitable for both binary and multiple-class categorization problems, and can be applied as a dimensionality reduction technique. In the latter case, the number of discriminative features is determined automatically since the process of feature extraction is fully embedded in the optimization procedure. Performance tests validate our method on a number of benchmark data sets from the UCI repository, while the experiments in the application of visual object and content-based image categorization demonstrate very competitive results, asserting the method's capability of producing semantically relevant matches that share the same or synonymous vocabulary with the query category and allowing multiple pertinent category assignment.

1. Introduction

Object categorization, as a fundamental computer vision problem, has long been a major focus of ongoing research, which lead to the development of a variety of methods and techniques proposed to date, e.g., [14, 15, 30]. Many approaches have demonstrated impressive results in specific aspects of the recognition, categorization and classification tasks, such as feature design [27] and extraction [23], known object detection [29], etc. However, there has been a certain lack of attention to the development of general methods that can achieve high recognition accuracy by actively transforming the data and reducing its dimensionality to extract features in an optimal fashion, while taking into account specific properties of the underlying classifier. With the exception of several notable contributions, e.g., [6, 13, 16, 33], many approaches essentially treat the classifier as a black box completely isolated from the feature extraction process, which ultimately leads to suboptimal results. There exist hundreds of dimensionality-reducing data transformation methods originating from families as diverse as discriminant analysis techniques (LDA, DF-LDA, GDA), their advanced extensions (SHOSLIF, Fisherfaces), nonlinear mappings (MDS, SOM) and neural networks (NeuroScale), referencing all of which is simply impossible due to their sheer number. Yet, the answers to important questions, such as "How many dimensions are enough to discriminate among given classes?", still remain vague.

In order to address these issues, we investigate the problem of object categorization in the discriminant analysis framework and propose a method of finding a distance-based discriminative transformation of the original visual feature data. Based on the compactness hypothesis [1], the sought transformation specifically aims at improving the accuracy of the nearest neighbor (NN) classifier [11] and implicitly integrates the feature extraction process in the problem formulation. Additional constraints are imposed to prevent overfitting and thus improve generalization abilities of the proposed method.

The remainder of this paper is structured as follows. In section 2 we formulate the task of deriving a discriminant transformation as a problem of minimizing an asymmetric criterion based on the compactness hypothesis. In section 3 we review the iterative majorization method and demonstrate how it can be used to minimize the chosen criterion. Section 4 provides a complete algorithm that obtains the sought transformation alongside with the extensions of the proposed approach for dimensionality reduction and multiple class discriminant analysis. We detail our experimental results for both benchmark and real-world image data sets in section 5 and discuss the proposed approach by highlighting its essential differences from existing methods in section 6.

2. Problem formulation

Suppose that we seek to distinguish between two classes represented by data sets X and Y having N_X and N_Y m-dimensional observations, respectively. For this purpose, we are looking for such transformation matrix[1] $T \in \mathbb{R}^{m \times m}$ such that $\{X \mapsto X', Y \mapsto Y'\}$, that places instances of a given class near each other while relocating the instances of the other class sufficiently far away. In other words, we want to ensure that the compactness hypothesis holds for either of the two classes in question, while its opposite is true for both.

Now, we must reiterate that our primary goal is to improve the NN performance on the task of discriminant analysis. This implies, first of all, that the sought problem formulation must relate only to the factors that directly influence the decisions made by the NN classifier, namely - the distances among observations. Secondly, in order to benefit as much as possible from the non-parametric nature of the NN, the sought formulation must not rely on the traditional class separability and scatter measures that use class means, weighted centroids or their variants (e.g., see [12]) which, in general, connote quite strong distributional assumptions. Finally, an asymmetric product form should be more preferable, since the sum of distances is inadequate as a between-class separability measure due to the fact that it can be made arbitrarily large while at the same time having a significant proportion of summand distances close to zero, while the asymmetry may be justified as consistent with the properties of the data encountered in the target application area of multimedia retrieval and categorization [35]. More formally, these requirements can be accommodated by an optimization criterion expressed in terms of distances among the observations from the two data sets as follows.

Let $d_{ij}^W(T)$ denote a Euclidean distance between points i and j *within* transformed data set X' given a transformation matrix T, and, analogously, $d_{ij}^B(T)$ specify a distance *between* the i-th point from data set X' and the j-th point from data set Y'. Using this notation, the sought discriminative data transformation can be obtained by minimizing the following criterion:

$$J(T) = \frac{\left(\prod_{i<j}^{N_X} \Psi\left(d_{ij}^W(T)\right) \right)^{\frac{2}{N_X(N_X-1)}}}{\left(\prod_{i=1}^{N_X} \prod_{j=1}^{N_Y} d_{ij}^B(T) \right)^{\frac{1}{N_X N_Y}}}, \qquad (1)$$

where the numerator and denominator of (1) represent the geometric means of corresponding distances, and $\Psi(\cdot)$ denotes a Huber robust estimation function [18]. The choice of Huber function in (1) is motivated by its ability to switch from quadratic to linear penalty allowing to mitigate the consequences of an implicit unimodality assumption that the formulation of the numerator of (1) leads to. Additionally, Huber function has several attractive properties that greatly facilitate the derivation of the majorizing inequalities, as will be shown in section 3.2.

In the logarithmic form, criterion (1) is written as:

$$\begin{aligned} \log J(T) &= \frac{2}{N_X(N_X-1)} \sum_{i<j}^{N_X} \log \Psi\left(d_{ij}^W(T)\right) \\ &\quad - \frac{1}{N_X N_Y} \sum_{i=1}^{N_X} \sum_{j=1}^{N_Y} \log d_{ij}^B(T) \qquad (2) \\ &= \alpha S_W(T) - \beta S_B(T). \end{aligned}$$

The first and the second summation terms of (2) are going to be referred to as $S_W(T)$ ("within" distances) and $S_B(T)$ ("between" distances) in the following discussion to allow for a more convenient notation and due to their *functional* similarity with the notions of within- and between-class scatter measures used in a number of well-known discriminant analysis techniques [9, 10, 13, 16]. We will also shorten the notation by reassigning the normalizing quantities $\frac{2}{N_X(N_X-1)}$ and $\frac{1}{N_X N_Y}$ to α and β, respectively.

Although a straightforward differentiation of (2) might be sufficient in order to proceed with a generic optimization search technique such as gradient descent, our preliminary experiments showed that computational costs of such an endeavor can very quickly become prohibitive, especially if one adheres strictly to the main premise of this work, i.e., uses only pairwise distances among observations (quadratic complexity), as opposed to deviations from class means (linear complexity) of the customary class separability and scatter measures abundant in clustering literature. The computational cost situation will be further exacerbated if, in addition to the descent direction, a proper step length must be calculated, so that gradient descent does not overshoot and actually manages to improve the optimization criterion, while the latter outcome is guaranteed by the introduced below iterative majorization technique (and, hence its alternative name: "guaranteed descent"). Furthermore, given the formulation of (2), some of the tested state-of-the-art optimization routines (SQP, Quasi-Newton with line search) happened not to be able to converge, even on fairly simple data sets.

Therefore, before considering possible strategies for optimizing (2), it is beneficial to derive some useful approximations of $S_W(T)$ and $S_B(T)$ that can make the task of minimizing $\log J(T)$ criterion amenable to a simple iterative procedure based on the majorization method, which we discuss in the following section.

[1] At the moment, we consider T to be a square matrix. Section 4.2 on dimensionality reduction will deal with T of size $m \times k$, where $k \ll m$.

3. Iterative majorization

3.1. General overview of the method

As stated in [5, 32, 17], the central idea of the majorization method is to replace the task of optimizing a complicated objective function $f(x)$ by an iterative sequence of simpler minimization problems in terms of the members of the family of auxiliary functions $\mu(x, \bar{x})$, where x and $\bar{x}$ vary in the same domain Ω. In order for $\mu(x, \bar{x})$ to qualify as a *majorizing function* of $f(x)$, the auxiliary function $\mu(x, \bar{x})$ is required to: (a) have a unique minimum, (b) always be greater than or equal to the original objective function, and (c) touch the surface of the original function at the *supporting point* $\bar{x}$.

Once an appropriate function $\mu(x, \bar{x})$ has been found, the iterative majorization algorithm proceeds as follows. After assigning an initial supporting point $\bar{x}$, the successor point x_s is found by minimizing $\mu(x, \bar{x})$. The obtained x_s subsequently becomes the next supporting point, and the process repeats until there is no improvement in the value of the objective function.

The essential property of the above procedure is that it generates a non-increasing sequence of function values, which converges to a stationary point whenever $f(x)$ is bounded from below and x is sufficiently restricted. In additional to its computational advantages, the majorization method has the valuable properties of low dependence on the initial value [32] and enhanced robustness with respect to local minima problems [20]. In the next section, we will outline a way to derive majorizing expressions of (2) and show how they can be used for optimizing the chosen criterion.

3.2. Majorizing the optimization criterion

It can be verified that majorization remains valid under additive decomposition [17]. Therefore, a possible strategy for majorizing (2) is to deal with $S_W(T)$ and $-S_B(T)$ separately and subsequently recombine their respective majorizing expressions.

We begin by noting that both logarithm and Huber distance are majorizable by linear and quadratic functions, respectively [17, 21]. This fact makes it possible to derive a majorizing function of $S_W(T)$ as follows:

$$\begin{aligned} S_W(T) &= \sum_{i<j}^{N_X} \log \Psi\left(d_{ij}^W(T)\right) \\ &\leq \sum_{i<j}^{N_X} \frac{\bar{w}_{ij} \cdot \left(d_{ij}^W(T)\right)^2}{2\Psi\left(d_{ij}^W(\bar{T})\right)} + K_1 \\ &= \mu_{S_W}(T, \bar{T}), \end{aligned} \tag{3}$$

where $T, \bar{T} \in \mathbb{R}^{m \times m}$, $\bar{T}$ is a supporting point for T, $\bar{w}_{ij}$ is a weight of the Huber function majorizer, as defined in [17], and K_1 is a constant that collects all of the terms that are irrelevant from the point of view of minimization with respect to T (see [21] for detailed derivations). In the matrix form, the above formulation can be expressed as:

$$\mu_{S_W}(T, \bar{T}) = \frac{1}{2}\mathbf{tr}\left(T^T X^T R X T\right) + K_1, \tag{4}$$

where R is a square symmetric design matrix, as specified in [21].

As for $-S_B(T)$, we start out by expressing its every term using a second order Taylor series expansion of the logarithm function around a supporting point $\bar{T}$. In the resulting formulation, the sum of Euclidean distances can be majorized by a rule based on the Cauchy-Schwarz inequality [5, 17] (again, see [21] for derivation details). In the matrix form, the resulting majorizing function of $-S_B(T)$ can be expressed as:

$$\begin{aligned} \mu_{-S_B}(T, \bar{T}) &= \frac{1}{2}\mathbf{tr}(T^T Z^T G Z T) \\ &\quad -2\mathbf{tr}(T^T Z^T G Z \bar{T}) + K_2, \end{aligned} \tag{5}$$

where Z is the matrix obtained by joining X and Y together, row-wise, and G is a square symmetric design matrix of size $N = N_X + N_Y$, as defined in [21].

Finally, combining results (4) and (5), we obtain a majorizing function of the $\log J(T)$ optimization criterion:

$$\begin{aligned} \mu_{\log J}(T, \bar{T}) &= \alpha\mu_{S_W} + \beta\mu_{-S_B} \\ &= \frac{\alpha}{2}\mathbf{tr}\left(T^T X^T R X T\right) \\ &\quad + \frac{\beta}{2}\mathbf{tr}(T^T Z^T G Z T) \\ &\quad -2\beta\mathbf{tr}(T^T Z^T G Z \bar{T}) + K_3, \end{aligned} \tag{6}$$

that can be used to find an optimal transformation T minimizing $\log J(T)$ criterion via the iterative procedure outlined in section 3.1. Similarly to the last terms in (4) and (5), K_3 is a constant that collects all of the other terms that are irrelevant from the point of view of minimization with respect to T.

3.3. Minimization of the majorizer of $\log J(T)$

It is possible to minimize (6) with respect to T in a straightforward fashion by setting its derivative to zero and solving the resulting system of linear equations. However, it is often recommended [2, 3, 22, 24] that a length-constrained solution be found by deploying such techniques as weight-limiting, weight decay, etc., especially in the case of classifiers capable of achieving zero training error, to

prevent overfitting and thus improve generalization performance of the classifier. By incorporating the constraint into a Lagrangian, solving it and substituting the solution back into the expression of the length constraint, we obtain the following:

$$\mathcal{E}_0 = \mathbf{tr}\left(L^T U \frac{1}{(2\lambda I + D)^2} U^T L\right), \tag{7}$$

where λ is a Langrangian multiplier, $\mathcal{E}_0$ is the value of the length constraint estimated from the validation data set, M is defined as $\frac{\alpha}{\beta} X^T R X + Z^T G Z$, L is equal to $2 Z^T G Z \bar{T}$, I is an identity matrix, and U, D are the respective matrices of eigenvectors and eigenvalues of M[2]. Clearly, (7) is easily solved by any suitable root-finding technique, such as Newton-Raphson method. Once the constraint-satisfying value λ has been found, the optimal transformation T, i.e. the successor point in the iterative majorization algorithm is recovered as:

$$T_s = U \frac{1}{2\lambda I + D} U^T L. \tag{8}$$

4. Putting it all together

4.1. Complete algorithm

Considering all of the derivations we have desribed so far, the complete distance-based discriminant analysis (DDA) algorithm for iterative majorization of $\log J(T)$ criterion (2) can be specified as follows:

1. Assign an initial supporting point $\bar{T} = \bar{T}_0 \in \mathbb{R}^{m \times m}$;
2. Find a successor point T_s using (8);
3. If $\log J(\bar{T}) - \log J(T_s) < \epsilon$, then stop;
4. Set $\bar{T} = T_s$, go to 2.

4.2. Dimensionality reduction

As previously mentioned in section 2, the initial choice of the size of the sought discriminative transformation was to have $T \in \mathbb{R}^{m \times m}$. However, so far we have not encountered a single statement that requires T to be a square matrix, which we can take advantage of by setting the column size of T equal to a certain $k \ll m$, rendering the presented method a dimensionality reduction technique as well.

Furthermore, the exact value of k is readily available, providing an answer to a critically important question as for how many dimentions of data one needs to retain in order to preserve the same discriminatory power as in the full-dimensional case. Indeed, since the distances among observations are the same in both TT^T and US^2U^T metrics (where U and S come from the singular value decomposition of T: $T = USV^T$), the effect of applying $T \in \mathbb{R}^{m \times m}$ is fully captured by the left-singular vectors of T scaled by non-zero singular values, whose number, in turn, gives the exact value of k. Some examples, as well as a summary of various properties that distinguish DDA from other dimensionality reduction methods are given in section 6.

4.3. Multiple class discriminant analysis

Another possible modification is a multiple-class extension of the DDA technique, expressed by the following definition of the optimization criterion:

$$\log J_K(T) = \sum_{i=1}^{K-1} \left(\alpha^{(i)} S_W(T)^{(i)} - \beta^{(i)} S_B(T)^{(i)}\right) \tag{9}$$

where K is the number of classes, $K \geq 2$. Note that (9) becomes (2) for the two-class formulation, when $K = 2$. Again, similarly to the latter case, the particular class to be left out (since (9) considers only $K - 1$ terms) may be determined using domain knowledge, or via statistical techniques. In order to accommodate the changes required for adopting (9), the individual matrices R and G from (4) and (5) will be replaced with

$$R_K = \sum_{i=1}^{K-1} \frac{\alpha^{(i)}}{\beta^{(i)}} R^{(i)}, \quad \text{and} \quad G_K = \sum_{i=1}^{K-1} G^{(i)}, \tag{10}$$

respectively, where each of the matrices $R^{(i)}$ is computed similarly to that of (4) using observations from class i, while matrices $G^{(i)}$ are calculated as used in (5) with proper index interval adjustment for computing distances between data points of a given class i and the rest of the data set.

5. Experimental results

5.1. Benchmark data set performance

In our preliminary empirical analysis, the error rate results of classification performance (on both binary and multiple-category data sets) of two types of experiments were compared. For the first type of experiments, which we will refer to as simply "NN" experiments, we measured classification error rate of the NN classifier using 10-fold cross-validation [34]. In the second type of experiments, that are going to be called "DDA+NN" experiments, an additional stage of applying a discriminating transformation

[2] Simplifying the notation of (7), the reciprocal and squaring operations should be understood as applied to the diagonal matrix D on the element by element basis taking into account the magnitudes of each eigenvalue so as to avoid division by zero problems.

T derived with the proposed DDA method prior to measuring the cross-validation performance of the NN classifier was introduced. Therefore, the goal of this analysis was to assess the effect of applying a DDA transformation on the accuracy of the NN classifier.

Several well-known data sets from the UCI Machine Learning Repository [4] were used in our experiments. The error rates of NN and DDA+NN data classification experiments averaged over twenty trial cross-validation runs are presented in Table 1, showing an improvement in performance and comparing favorably with the best results on these data sets published to date [19].

Table 1. Classification results for UCI data

Data set	Classes	% Err. NN	% Err. DDA+NN
Hepatitis	2	29.57	0.00
Ionosphere	2	13.56	7.73
Diabetes	2	30.39	27.11
Heart	2	40.74	21.11
Monk's P1	2	14.58	0.69
Balance	3	21.45	3.06
Iris	3	4.00	3.33
DNA	3	23.86	6.07
Vehicle	4	35.58	24.70

5.2. Application to visual object categorization

For our object categorization experiments we chose a recently developed database ETH80 composed of entities corresponding to the basic level of human knowledge organization [25]. The database contains high-resolution color images of 80 objects from 8 different classes, for a total of 3280 images. The visual information for each image was represented by 286-dimensional feature vector containing 166 global color histogram and 120 Gabor filter texture descriptors extracted by the *Viper* system [31]. The training set comprised images taken one per class object viewed from a fixed position, while the rest was allocated to the test set. Again, similarly to the setup described above (see section 5.1), we compared performance results for "NN"and "DDA+NN" experiments for each of the 8 classes, but this time, using a one-against-all classification configuration typically encountered in ensemble learning [8], and setting target dimensionality to 2D. The results are summarized in Table 2.

It is importnant to emphasize here that image representation for these experiments was reduced via DDA to two dimensions only. Nevertheless, as shown in the last column of Table 2, the proposed technique still was able to descrease categorization error rate, which improved the overall performance average. The results in Table 2 also reveal the importance of the length constraint for the purpose of avoiding data over-fitting problems. Both unconstrained and constrained solutions found by the DDA procedure lead to zero error rate on the training data, but, as can be easily seen from Table 2, their performance turned out to be drastically different on the test data sets, demonstrating an adequate generalization capability induced by the length-constrained version of the proposed method. An example of the 2D representation of the training set for image class 2 obtained by DDA is shown in Figure 1. As can be easily seen from the figure, the target class images are well separated from those of all of the other classes, which is exactly the requirement one seeks to satisfy in one-against-all classification.

Table 2. Object categorization results for the ETH80 image database

Object class	% Error rate		
	NN	DDA+NN (unconstrained)	DDA+NN (constrained)
(1) Apple	4.47	18.66	0.75
(2) Car	14.47	18.72	5.78
(3) Cow	12.12	16.91	10.97
(4) Cup	3.09	16.94	2.22
(5) Dog	14.00	16.66	12.72
(6) Horse	14.47	14.84	13.16
(7) Pear	6.13	18.94	3.84
(8) Tomato	2.50	16.87	1.88

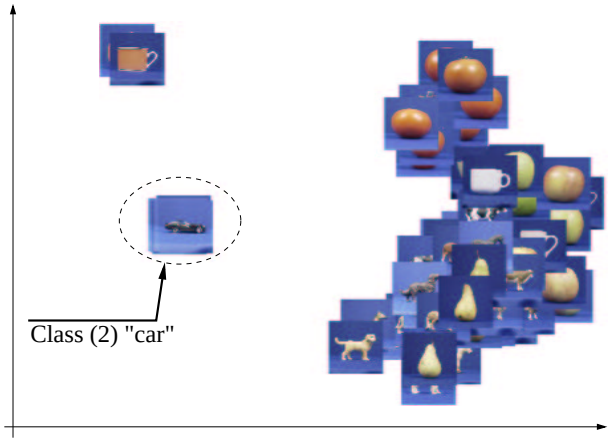


Figure 1. Result of applying a dimensionality-reducing DDA transformation to the training set for class (2). Images from class 2 are projected close to each other while images belonging to the other classes are freely scattered maintaining a certain distance margin from class 2

In addition to the tests mentioned above, we also ex-

Figure 2. Categorization of previously unseen images. The annotation keywords overlapping with the query category vocabulary are listed in bold font.

plored empirically the influence of the DDA transformation on the performance of other classification methods (including NN as a baseline) on the real-world image categorization. For these experiments, three (potentially overlapping) image sets were selected from the Washington University annotated image collection [26], based on the presence of keywords "trees", "cars" and "ocean" in their annotation. Every classifier was then tested by 10-fold cross-validation on the task of categorization of test images as belonging to a given class, e.g., "trees", based on their 286-dimensional visual feature vector representation (see above). The remarkable results of these experiments demonstrate that applying the DDA transformation not only consistenly improves NN classifier accuracy (as expected), but also provides a boost in performance to some more advanced non-linear classification methods, such as SVM [7], as shown in Table 3.

Table 3. Image categorization results

Classifier	% Error on image data set		
	Trees	Ocean	Cars
Fisher's LDA	43.89	45.56	17.72
SVM (linear)	31.11	21.11	1.58
DDA+SVM (linear)	17.78	11.11	1.40
SVM (gaussian)	23.89	16.67	1.58
DDA+SVM (gaussian)	17.78	11.11	1.40
NN	38.33	19.44	2.46
DDA+NN	18.89	18.33	1.23

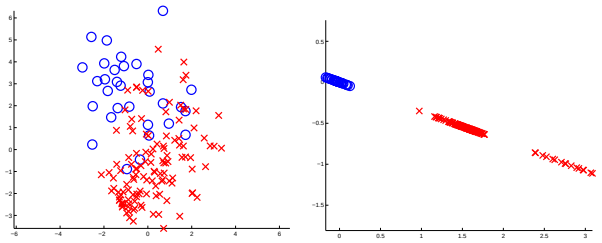
In order to verify that non-trivial collection-independent learning has occurred, we also examined the categorization performance of the derived above category-specific DDA transformations on a completely separate image set taken from the COREL database. The empirical evidence demonstrates that the application of the DDA transformation leads to robust categorization of unseen images producing semantically relevant matches that may (Figure 2, row one) or may not (Figure 2, row two) share the same vocabulary with the query category, as well as allowing images to be assigned to multiple relevant categories (Figure 2, the last two images in both rows).

6. Discussion and related work

In this section we briefly review some of the previously developed approaches of discriminant analysis and dimensionality reduction, demonstrating on simple examples the essential differences between existing techniques and the proposed DDA method.

First, we consider principal component analysis (PCA), a fundamental tool for dimensionality reduction that finds a set of orthogonal vectors that account for as much as possible of the data's variance. Apparently, PCA method disregards class membership information altogether and consequently is of limited use as a discriminatory transform.This conjecture is easily confirmed by comparing 2D projections of the Hepatitis data set by the PCA and DDA methods illustrated in Figure 3, which shows a perfect class separation for the latter approach explaining its 100% classification accuracy reported earlier. The singular value decomposition of the resulting transformation reveals that there is only one significantly different from zero singular value, meaning that in order to distinguish between the two classes one may use just one dimension, i.e., project the data set onto a line, as seen in Figure 3(b).

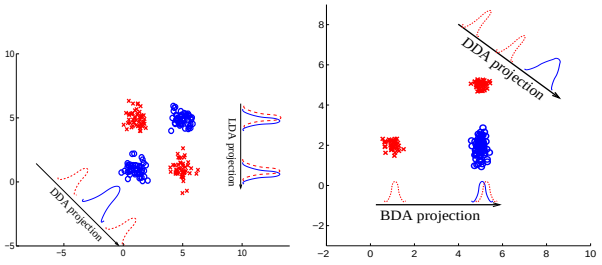
Fisher's linear discriminant analysis (LDA) [9, 10, 12] projects original data into a smaller number of dimensions, while trying to preserve as much disriminatory information as possible by maximizing the ratio of between-class



(a) PCA projection (b) DDA projection

Figure 3. 2D projections of the Hepatitis data

scatter over within-class scatter. Based on the second order statistical information, the method is proven to be optimal whenever data classes are represented by unimodal Gaussians with well-separated means. A violation of this assumption drastically deteriorates LDA's performance, as seen in Figure 4(a) that compares discriminative projections found by LDA and DDA methods for the classical XOR problem [9]. As for the DDA approach, Figure 4(a) illustrates that the proposed technique does not require data Gaussianity assumptions. Furthermore, the method can determine discriminative projection transformations of up to as many dimensions as there are in the data (even though it is hardly ever necessary to retain more than k, see section 4.2), whereas LDA is limited by rank restrictions on the between-class scatter matrices to have no more than $K-1$ dimensions, where K is the number of classes.



(a) XOR problem solved by the LDA and DDA

(b) "Dominant variance direction" problem solved by the BDA and DDA

Figure 4. Comparison of LDA, BDA and DDA

A biased discriminant analysis (BDA) approach [35] developed with a goal in mind to improve efficiency of interactive multimedia retrieval applications, is, similarly to the DDA, based on an appealing idea of asymmetric treatment of positive and negative relevance feedback examples. This technique excels in overcoming several important drawbacks of parametric approaches, such as LDA and MDA [9], induced by scatter matrix rank restrictions and Gaussianity assumptions and, conceptually, is closest to the two-class version of the proposed DDA method. However BDA's performance is offset by suboptimal solutions whenever the observations from the two classes overlap considerably along the direction orthogonal to that of minimal variance of the positive examples (see Figure 4(b)).

There also exist other DA methods that are specifically designed to work well for non-Gaussian data sets (e.g., NDA [13]) and target the nearest neighbor classifier performance (e.g., a recent enhancement of NDA proposed in [6]), whose main difference from DDA lies in the fact that these methods still rely on parametric within-class scatter matrices. Among the iterative techniques, there is discriminant adaptive nearest neighbor (DANN) approach [16] that relies on a probabilistic formulation to achieve similar goals, and the class-dependent weighted (CDW) dissimilarity method [28] that can effectively be considered operating in a restricted case of the DDA setting where no feature extraction is possible as the sought transformation is required to be a diagonal matrix.

7. Conclusion

We have described a visual object categorization method formulated in the discriminant analysis framework. The main focus of the proposed approach is on finding a transformation of the original data that enhances the degree of conformance to the compactness hypothesis and its inverse, which has been shown to lead to an improved categorization accuracy.

The presented method can be used as a dimensionality reduction technique with an advantageous ability to determine automatically the number of necessary discriminative features, is suitable for both binary and multiple-class categorization problems, and preserves non-parametric properties of the underlying classifier by depending exclusively on the inter-observation distances (hence, the name of the approach).

The performance of the proposed method has been verified on a number of the benchmark data sets, and tested on the visual object categorization tasks. The encouraging results not only demonstrated DDA's superiority compared to a number of baseline techniques, but also revealed that the method improves the results achieved by some more advanced non-linear classification methods, such as SVM.

Acknowledgements

This work is funded by Swiss National Fundation (NCCR-IM2 and research grant 21-66648-01) and EU-IST project Webkit (FP5).

References

[1] A. Arkadev and E. Braverman. *Computers and Patter Recognition*. Thompson, Washington, D.C., 1966.

[2] P. Bartlett. For valid generalization, the size of the weights is more important than the size of the network. In *Advances in Neural Information Processing Systems 9*, pages 134–140, 1997.

[3] M. Bertero and P. Boccacci. *Introduction to Inverse Problems in Imaging*. Institute of Physics Publishing, January 1998.

[4] C. Blake and C. Merz. UCI repository of machine learning databases, 1998.

[5] I. Borg and P. J. F. Groenen. *Modern Multidimensional Scaling*. New York, Springer, 1997.

[6] M. Bressan and J. Vitri. Nonparametric discriminant analysis and nearest neighbor classification. *Pattern Recognition Letters*, 24(15):2743–2749, 2003.

[7] N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[8] T. G. Dietterich. Ensemble methods in machine learning. In J. Kittler and F. Roli, editors, *First International Workshop on Multiple Classifier Systems*, pages 1–15. Springer-Verlag, 2000.

[9] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley, June 1973.

[10] R. A. Fisher. The use of multiple measures in taxonomic problems. *Ann. Eugenics*, 7:179–188, 1936.

[11] E. Fix and J. Hodges. Discriminatory analysis: Nonparametric discrimination: Consistency properties. Technical Report 4, USAF School of Aviation Medicine, February 1951.

[12] K. Fukunaga. *Introduction to statistical pattern recognition*. Academic Press, New York, 2nd edition, 1990.

[13] K. Fukunaga and J. Mantock. Nonparametric discriminant analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (5):671–678, 1983.

[14] B. Funt and G. Finlayson. Color constant color indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):522–529, May 1995.

[15] T. Gevers and A. Smeulders. Color-based object recognition. *Pattern Recognition*, 32(3):453–464, March 1999.

[16] T. Hastie and R. Tibshirani. Discriminant adaptive nearest neighbor classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(6):607–616, 1996.

[17] W. Heiser. Convergent computation by iterative majorization: Theory and applications in multidimensional data analysis. *Recent advances in descriptive multivariate analysis*, pages 157–189, 1995.

[18] P. Huber. Robust estimation of a location parameter. *Annals of Mathematical Statistics*, 35:73–101, 1964.

[19] E. M. Kleinberg. A mathematically rigorous foundation for supervised learning. In *LNCS, Proc. of the First International Workshop on Multiple Classifier Systems*, volume 1857, pages 67–76, Caligari, Italy, June 2000. Springer-Verlag.

[20] S. Koisnov, S. Marchand-Maillet, and T. Pun. Iterative majorization approach to the distance-based discriminant analysis. Presented by S. Kosinov at "Conference of the GfKl 2004", Dortmund, Germany, March 9–11 2004.

[21] S. Kosinov. Visual object recognition using distance-based discriminant analysis. Technical Report 03.07, Computer Vision and Multimedia Laboratory, Computing Centre, University of Geneva, Rue Général Dufour, 24, CH-1211, Geneva, Switzerland, 2003.

[22] A. Krogh and J. A. Hertz. A simple weight decay can improve generalization. In J. E. Moody, S. J. Hanson, and R. P. Lippmann, editors, *Advances in Neural Information Processing Systems*, volume 4, pages 950–957. Morgan Kaufmann Publishers, Inc., 1992.

[23] T. Krüger, J. Wickel, P. Alvarado, and K.-F. Kraiss. Feature extraction from vrml models for view-based object recognition. *Proc. of the 4th European Workshop on Image Analysis for Multimedia Interactive Services "WIAMIS 2003"*, pages 391–394, 2003.

[24] S. Lawrence and C. Giles. Overfitting and neural networks: Conjugate gradient and backpropagation. In *Proceedings of the IEEE International Conference on Neural Networks*, pages 114–119. IEEE Press, 2000.

[25] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *International Conference on Computer Vision and Pattern Recognition (CVPR'03)*, pages 409–415, Madison, Wisconsin, June 2003.

[26] Y. Li and L. G. Shapiro. Object recognition for content-based image retrieval. In *Lecture Notes in Computer Science*. Springer-Verlag, to appear, 2004.

[27] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Conference on Computer Vision ICCV, Corfu*, pages 1150–1157, 1999.

[28] R. Paredes and E. Vidal. A class-dependent weighted dissimilarity measure for nearest neighbor classification problems. *Pattern Recognition Letters*, 21(12):1027–1036, 2000.

[29] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.

[30] B. Schiele and J. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–50, January 2000.

[31] D. M. Squire, W. Müller, H. Müller, and J. Raki. Content-based query of image databases, inspirations from text retrieval: inverted files, frequency-based weights and relevance feedback. In *The 11th Scandinavian Conference on Image Analysis*, pages 143–149, Kangerlussuaq, Greenland, june 1999.

[32] K. van Deun and P. J. F. Groenen. Majorization algorithms for inspecting circles, ellipses, squares, rectangles, and rhombi. Technical report, Econometric Institute Report EI 2003-35, 2003.

[33] H. Watanabe, T. Yamaguchi, and S. Katagiri. Discriminative metric design for robust pattern recognition. *IEEE Trans. Signal Processing*, 45(11):2655–2661, 1997.

[34] S. Weiss and C. Kulikowski. *Computer Systems That Learn*. Morgan Kaufmann, 1991.

[35] X. Zhou and T. Huang. Small sample learning during multimedia retrieval using BiasMap. In *IEEE Computer Vision and Pattern Recognition (CVPR'01)*, Hawaii, 2001.