

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Nearest Feature Rules and Dissimilarity Representations for Face Recognition Problems

Mauricio Orozco-Alzate and Germán Castellanos-Domínguez
*Universidad Nacional de Colombia Sede Manizales
 Colombia*

1. Introduction

Over the last decade, face recognition has been a widely-studied area of research. It has been mainly motivated by a high and always increasing demand of reliable authentication and security systems, as well as by numerous medical-related and human-computer interaction applications; such as posture/gesture recognizers, intelligent multimodal systems and speech therapy software. In addition, a variety of dimensionality reduction techniques and classification rules have been treated. In particular, linear transformations for extracting non-facial or non-geometric features and non-parametric pattern classifiers have been widely employed in the so-called *pixel-based approach*, which consists in operating directly on the acquired image, without deriving facial features such as the distance between eyes or the area of the mouth.

Face recognition is a particular problem of multi-class classification. In general, we are given a set of training objects $X := \{(\mathbf{x}_i, \omega_i) \mid \mathbf{x}_i \in \mathbf{R}^M, i = 1, \dots, N\}$, each of them (pixels from a face image in our particular case) consisting of a M dimensional pattern \mathbf{x}_i and its label $\omega_i \in \omega$. In pixel-based face recognition problems, feature extraction and feature selection methods are applied in order to reduce the dimensionality. Such methods usually consist in a transformation $\phi: X \rightarrow Z$, such that $\mathbf{z} := \phi(\mathbf{x})$.

The eigenface representation is the simplest and widest used dimensionality reduction technique employed in pixel-based face recognition. It consists in the principal component analysis (PCA) or the Karhunen-Loève transform (KL), differing mainly at the structure of the covariance matrix. Let \mathbf{x} be a vector formed by all the rows of an image, the prototype faces are arranged on a matrix $\bar{\mathbf{X}} = [\mathbf{x}_1 - \boldsymbol{\mu}_x \cdots \mathbf{x}_N - \boldsymbol{\mu}_x]$, where $\boldsymbol{\mu}_x = E[\mathbf{x}] = 1/N \sum_{n=1}^N \mathbf{x}_n$. Due to the fact that the number of training faces N is often smaller than the face dimension d , it is more advisable to calculate the eigenvectors of the $N \times N$ covariance matrix $\boldsymbol{\Sigma}'_{\bar{\mathbf{X}}} = \bar{\mathbf{X}}^T \bar{\mathbf{X}}$, instead of those of the $d \times d$ covariance matrix $\boldsymbol{\Sigma}_{\bar{\mathbf{X}}} = \bar{\mathbf{X}} \bar{\mathbf{X}}^T$. The eigenvectors \mathbf{w}_i corresponding to the p largest eigenvalues are called *eigenfaces* and determine a transformation matrix $\mathbf{W}_{\text{eigen}} = [\mathbf{w}_1 \mathbf{w}_2 \cdots \mathbf{w}_{N_p}]$, where $N_p \leq N$ is the number of principal components to be considered in further procedures. A specific value for N_p is selected according to some

criterion, e.g. the information percentage on the eigenvalues. A feature point is transformed by

$$\mathbf{z} = \mathbf{W}_{\text{eigen}}^T (\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}}) \quad (1)$$

This chapter is concerned with four subjects. The first one is a conceptual and experimental review of the nearest feature classifiers; some bibliographical remarks as well as theoretical and empirical conclusions are given. The second subject is a quantification of the computational complexity of the nearest feature rules, by using an economic model which takes into account a trade-off between classifier error and evaluation complexity. Complexities of these classifiers are estimated in terms of orders (big-oh notation) and measured in FLOPs. The study includes error-complexity curves and complexity costs, resembling a cost-benefit analysis. The third one corresponds to a face recognition task based on dissimilarity representations, which shows that normal density-based (Bayesian) classifiers constructed on such representations are an alternative approach to the direct application of the nearest neighbor rule. The last subject is aimed to present a conceptual discussion on the relationship between the nearest feature rules and dissimilarity representations, particularly the so-called *generalized dissimilarity representations* and their potential application to face recognition problems as well as to other applications. Some open and apparently promising issues to be considered for further research are also discussed in the concluding section.

2. The nearest feature classifiers

In classification theory, there is an approach completely independent of statistical knowledge or assumptions, the so-called *distribution free* classification, often referred to as *nonparametric* techniques. Such an approach includes classification algorithms which can be described without reference to probability distributions; i.e. without the assumption that the forms of the underlying densities are known (Duda et al., 2000).

Nonparametric procedures can be roughly divided into two branches: firstly, methods for estimating the underlying density functions, including the Parzen-window method and the k_n -nearest neighbor estimation; secondly, procedures for estimating directly the a posteriori probabilities such as the well-known k -nearest neighbor rule (k -NN) which, in spite of its simplicity, has been successfully used in a considerable variety of applications. Nonetheless, it requires a significant amount of storage and computational effort; such a problem can be partly solved by using the condensed nearest neighbor rule (CNN) (Hart, 1968). In addition, the k -NN classifier suffers of a potential loss of accuracy when a small set of prototypes is available. To overcome this shortcoming, the nearest feature classifiers were developed. They are also a type of nonparametric techniques, which are based on a measure of distance between the query point and the prototypes or a function calculated from them, such as a line, a plane or a space. In this work, we consider four different nearest feature rules: k -nearest-neighbor or k -NN, k -nearest-feature-line or k -NFL, k -nearest-feature-plane or k -NFP and nearest-feature-space or NFS. The two last ones were proposed in (Chien & Wu, 2002) as a complete geometric generalization of k -NFL.

Before defining the nearest feature classifiers, a brief comment on notation is given. Consider a collection of training faces $Z := \{\mathbf{z}_i, \omega_i\} | \mathbf{z}_i \in \mathbf{R}^N, i = 1, \dots, C \cdot n_c\}$, where C denotes the number of classes and n_c the number of objects per class. We assume, without

loss of generality, a transformed point \mathbf{z} because a dimensionality reduction technique is usually applied before using a classifier; however, for the sake of notation simplicity, \mathbf{x} and \mathbf{z} will be used indistinctly to denote a pattern. The nearest feature rules are defined as follows.

2.1 The k-Nearest-Neighbor Rule

The simplest nonparametric method for classification should be considered k-NN (Cover & Hart, 1967). This rule classifies \mathbf{z} by assigning it the class label \hat{c} most frequently represented among the k nearest prototypes; i.e., by finding the k neighbors with the minimum distances between \mathbf{z} and all prototype feature points $\{\mathbf{z}_{ci}, 1 \leq c \leq C, 1 \leq i \leq n_c\}$. For $k=1$, the rule can be written as follows:

$$d(\mathbf{z}, \mathbf{z}_{\hat{c}}) = \min_{1 \leq c \leq C; 1 \leq i \leq n_c} d(\mathbf{z}, \mathbf{z}_{ci}), \quad (2)$$

where $d(\mathbf{z}, \mathbf{z}_{ci}) = \|\mathbf{z} - \mathbf{z}_{ci}\|$ is usually the Euclidean norm. In this case, the number of distance calculations is $n = \sum_{c=1}^C n_c$.

2.2 The k-Nearest-Feature-Line

The *k-nearest-feature-line* rule, or k-NFL (Li & Lu, 1999), is an extension of the k-NN classifier. This method generalizes each pair of prototype feature points belonging to the same class, $\{\mathbf{z}_{ci}, \mathbf{z}_{cj}\}$ by a linear function L_{ij}^c , which is called the *feature line* (see Figure 1). The line is expressed by the span $L_{ij}^c = \text{sp}(\mathbf{z}_{ci}, \mathbf{z}_{cj})$. The query \mathbf{z} is projected onto L_{ij}^c as a point \mathbf{p}_{ij}^c . This projection is computed as

$$\mathbf{p}_{ij}^c = \mathbf{z}_{ci} + \tau(\mathbf{z}, \mathbf{z}_{cj} - \mathbf{z}_{ci}), \quad (3)$$

where $\tau = (\mathbf{z} - \mathbf{z}_{ci})(\mathbf{z}_{cj} - \mathbf{z}_{ci}) / \|\mathbf{z}_{cj} - \mathbf{z}_{ci}\|^2$, which is called the *position parameter*. The classification of \mathbf{z} is done by assigning it the class label \hat{c} most frequently represented among the k nearest feature lines, for $k=1$ that means:

$$d(\mathbf{z}, L_{ij}^{\hat{c}}) = \min_{1 \leq c \leq C; 1 \leq i, j \leq n_c; i \neq j} d(\mathbf{z}, L_{ij}^c), \quad (4)$$

where $d(\mathbf{z}, L_{ij}^c) = \|\mathbf{z} - \mathbf{p}_{ij}^c\|$. In this case, the number of distance calculations is $n_L = \sum_{c=1}^C n_c(n_c - 1)/2$.

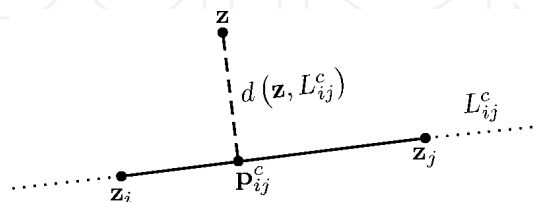


Figure 1. Feature line and projection point onto it

2.3 The k-Nearest-Feature-Plane

The *k-nearest-feature-plane* rule, or k-NFP, is an extension of the k-NFL classifier. This classifier assumes that at least three linearly independent prototype points are available for each class. It generalizes three feature points $\{z_{ci}, z_{cj}, z_{cm}\}$ of the same class by a feature plane F_{ijm}^c (see Figure 2); which is expressed by the span $F_{ijm}^c = \text{sp}(z_{ci}, z_{cj}, z_{cm})$. The query z is projected onto F_{ijm}^c as a point p_{ijm}^c . The projection point can be calculated as follows:

$$p_{ijm}^c = Z_{ijm}^c \left(Z_{ijm}^{cT} Z_{ijm}^c \right)^{-1} Z_{ijm}^{cT} z, \quad (5)$$

where $Z_{ijm}^c = [z_{ci} \ z_{cj} \ z_{cm}]$. Considering $k=1$, the query point z is classified by assigning it the class label \hat{c} , according to

$$d(z, F_{ijm}^{\hat{c}}) = \min_{1 \leq c \leq C; 1 \leq i, j, m \leq n_c; i \neq j \neq m} d(z, F_{ijm}^c), \quad (6)$$

where $d(z, F_{ijm}^c) = \|z - p_{ijm}^c\|$. In this case, the number of distance calculations is

$$n_F = \sum_{c=1}^C n_c(n_c - 1)(n_c - 2)/6.$$

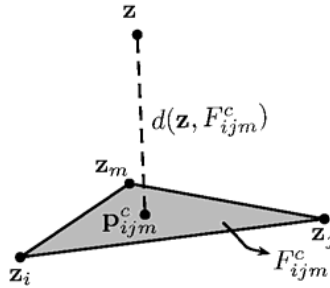


Figure 2. Feature plane and projection point onto it

2.4 The Nearest-Feature-Space Rule

The *nearest-feature-space* rule, or NFS, extends the geometrical concept of k-NFP classifier. It generalizes the independent prototypes belonging to the same class by a feature space $S^c = \text{sp}(z_{c1}, z_{c2}, \dots, z_{cn_c})$. The query point z is projected onto the C spaces as follows:

$$p^c = Z^c \left(Z^{cT} Z^c \right)^{-1} Z^{cT} z, \quad (7)$$

where $Z^c = [z_{c1} \ z_{c2} \ \dots \ z_{cn_c}]$. The query point z is classified by assigning it the class label \hat{c} , according to

$$d(z, S^{\hat{c}}) = \min_{1 \leq c \leq C} d(z, S^c) = \min_{1 \leq c \leq C} \|z - p^c\| \quad (8)$$

The number of distance calculations is always equals to C .

2.5 Theoretical geometric differences

It was geometrically shown in (Chien & Wu, 2002) that the distance of \mathbf{z} to F_{ijm}^c is smaller than that to the feature line. Moreover, the distance to the feature line is nearer compared with the distance to two prototype feature points. This relation can be written as follows:

$$d(\mathbf{z}, F_{ijm}^c) \leq \min(d(\mathbf{z}, L_{ij}^c), d(\mathbf{z}, L_{jm}^c), d(\mathbf{z}, L_{mi}^c)) \leq \min(d(\mathbf{z}, \mathbf{z}_{ci}), d(\mathbf{z}, \mathbf{z}_{cj}), d(\mathbf{z}, \mathbf{z}_{cm})) \quad (9)$$

In addition,

$$d(\mathbf{z}, S^c) = \min_{1 \leq c \leq C} d(\mathbf{z}, F_{ijm}^c) \quad (10)$$

In consequence, k-NFL classifier is supposed to capture more variations than k-NN, k-NFP should handle more variations of each class than k-NFL and NFS should capture more variations than k-NFP. So, it is expected that k-NFL performs better than k-NN, k-NFP is more accurate than k-NFL and NFS outperforms k-NFP.

2.6 Asymptotic behavior of the nearest feature rules

The problem of determining the error bound for the nearest feature rules can be addressed following the procedure to derive the error rate for the nearest neighbor rule; i.e. k-NN for $k=1$. The nearest feature rules are sub-optimal procedures as the k-NN rule; that is, they lead to an error rate greater than the minimum possible, the Bayes rate (Duda et al., 2000). In particular, for the k-NN rule with an unlimited number of prototypes, the error rate is never worse than twice the Bayes rate.

In this sense, the infinite-sample conditional average probability of error $P(e|x)$ and the unconditional average probability of error $P(e)$ are analyzed to find their minimum possible values: $P^*(e|x)$ and $P^*(e)$ respectively. Values of $P(e|x)$ and $P(e)$ are related, through the density $p(x)$, by

$$P(e) = \int P(e|x)p(x)dx. \quad (11)$$

Let us define the m -th state of nature $\omega_m(x)$ by $P(\omega_m|x) = \max_i P(\omega_i|x)$. The probability of error is minimized by the Bayes' decision rule, minimizing $P(e|x)$ for every x , thus

$$P^*(e|x) = 1 - P(\omega_m|x), \quad (12)$$

and

$$P^* = \int P^*(e|x)p(x)dx. \quad (13)$$

Expression (13) is called the *Bayes rate*.

A conditional probability of error $P(e|x, x')$ must be defined because the nearest neighbor rule depends on the samples, particularly on both the nearest prototype x' to a test point x and on the point x itself. $P(e|x)$ is obtained by averaging over x'

$$P(e|x) = \int P(e|x, x')p(x'|x)dx'. \quad (14)$$

In order to simplify the analysis of (14), the infinite-sample case, i. e. when n goes to infinity, is considered. In those conditions, the conditional density $p(x'|x)$ approaches to a delta function centered at x : $p(x'|x) \rightarrow \delta(x'-x)$ (See also (Duda et al., 2000) for a detailed demonstration). Now, an expression for $P(e|x, x')$ is derived as follows:

Let $(x_1, \theta_1), (x_2, \theta_2), \dots, (x_n, \theta_n)$ be n independently drawn labelled samples, where $\theta_j \in \{\omega_1, \dots, \omega_c\}$ for $j=1, \dots, n$. Suppose that a test point (x, θ) and its nearest training sample (x_j', θ_j') are selected. Since the states of nature, when x and x_j' were drawn, are independent, we have

$$P(\theta, \theta_j' | x, x_j') = P(\theta | x)P(\theta_j' | x_j'); \quad (15)$$

according to the nearest neighbor rule, an error is made if $\theta \neq \theta_j'$ and, consequently, the conditional probability of error $P_n(e|x, x_j')$ is given by

$$\begin{aligned} P_n(e|x, x_j') &= 1 - \sum_{i=1}^c P(\theta = \omega_i, \theta' = \omega_i | x, x_j') \\ &= 1 - \sum_{i=1}^c P(\omega_i | x)P(\omega_i | x_j'). \end{aligned} \quad (16)$$

Substituting (16) in (14):

$$\begin{aligned} \lim_{n \rightarrow \infty} P_n(e|x) &= \int \left[1 - \sum_{i=1}^c P(\omega_i | x)P(\omega_i | x') \right] \delta(x'-x) dx' \\ &= 1 - \sum_{i=1}^c P^2(\omega_i | x) \end{aligned} \quad (17)$$

In addition, if $P = \lim_{n \rightarrow \infty} P_n(e)$ and using (11) and (17) we have

$$\begin{aligned} P &= \lim_{n \rightarrow \infty} P_n(e) \\ &= \lim_{n \rightarrow \infty} \int P_n(e|x)p(x)dx \\ &= \int \lim_{n \rightarrow \infty} P_n(e|x)p(x)dx \\ &= \int \left[1 - \sum_{i=1}^c P^2(\omega_i | x) \right] p(x)dx \end{aligned} \quad (18)$$

Comparing (13) and (18), it can easily be seen that P^* is a lower bound on P . In order to calculate an upper bound, expression $\sum_{i=1}^c P^2(\omega_i | x)$ in (18) is examined to determine how it is minimized. Such an expression can be rewritten as

$$\sum_{i=1}^c P^2(\omega_i | x) = P^2(\omega_m | x) + \sum_{i \neq m} P^2(\omega_i | x), \quad (19)$$

and the bound for $\sum_{i=1}^c P^2(\omega_i | x)$ is found by minimizing the term $\sum_{i \neq m} P^2(\omega_i | x)$, s.t.:

$$P(\omega_i | x) \geq 0 \quad (20)$$

$$\sum_{i \neq m} P(\omega_i | x) = 1 - P(\omega_m | x) = P^*(e | x) \quad (21)$$

$\sum_{i=1}^c P^2(\omega_i | x)$ is minimized if $P(\omega_i | x) = P(\omega_j | x), \forall i, j \neq m$. Besides, from (21) we have:

$$P(\omega_i | x) = \begin{cases} \frac{P^*(e | x)}{c-1} & i \neq m \\ 1 - P^*(e | x) & i = m \end{cases} \quad (22)$$

The following inequalities can be derived from the expressions above:

$$\sum_{i=1}^c P^2(\omega_i | x) \geq (1 - P^*(e | x))^2 + \frac{P^{*2}(e | x)}{c-1} \quad (23)$$

and

$$1 - \sum_{i=1}^c P^2(\omega_i | x) \leq 2P^*(e | x) - \frac{c}{c-1} P^{*2}(e | x) \quad (24)$$

By substituting (24) in (18), it can be seen that $P \leq 2P^*$. Furthermore, a tight expression can be obtained by observing the variance of $P^*(e | x)$ (Duda et al., 2000):

$$\begin{aligned} \text{var}[P^*(e | x)] &= \int [P^*(e | x) - P^*]^2 p(x) dx \\ &= \int P^{*2}(e | x) p(x) dx - P^{*2} \geq 0 \end{aligned} \quad (25)$$

and, in consequence,

$$\int P^{*2}(e | x) p(x) dx \geq P^{*2} \quad (26)$$

Using (24) and (26) in (18), we obtain the inequality:

$$P^* \leq P \leq P^* \left(2 - \frac{c}{c-1} P^* \right), \quad (27)$$

which shows that the nearest neighbour error rate P in a multi-class (c classes) problem, having an infinite collection of training data, is always less than or equal to twice the Bayes rate. An elegant conclusion from (27) is given in (Duda et al., 2000): "at least half of the classification information in an infinite data set resides in the nearest neighbor".

Having an arbitrarily large number of prototypes, training or representation sets are fully informative and representative of the underlying processes. Since the nearest feature rules

attempt to enrich the representation and, under the condition cited above, available prototypes are fully representative, we intuitively do not expect a difference between the asymptotic behavior of the k -NN rule and the asymptotic behavior of the nearest feature classifiers for the infinite-sample case. The finite-sample case cannot be addressed by using such a simple reasoning. In fact, questions such as how rapidly the performance converges to the asymptotic value have still not been solved for the k -NN rule (Duda et al., 2000).

3. Quantifying the Computational Complexity of the Nearest Feature Classifiers

This section is devoted to quantifying the computational complexity of the nearest feature classifiers, by using an economic model which takes into account a trade-off between classifier error and evaluation complexity. The model is applied to the face recognition problem, which is the framework where these classifiers were originally proposed. Classifiers are also studied by measuring them in orders, denoted by the Landau symbol O (big-oh notation).

3.1 Complexity of the Nearest Feature Classifiers

Due to, as mentioned above, the nearest feature classifiers are non-parametric, the number of samples in the training set (prototypes) has a strong influence on the evaluation complexity. Since in modern computer systems, additions and multiplications are comparable in complexity (de Ridder et al., 2002), we can consider that:

- sum of two d -dimensional vectors costs d additions, therefore it has a complexity of d ,
- multiplication of two d -dimensional vectors costs d additions and d multiplications, so it has a complexity of $2d$,
- a scalar-vector multiplication has a complexity of d ,
- multiplying a $m \times d$ matrix by a $d \times 1$ vector has a complexity of $2dm$,
- multiplying a $m \times d$ matrix by a $d \times m$ matrix has a complexity of $m^2(2d-1)$,
- a $m \times m$ matrix inversion has a complexity of $O(m^3)$.

Considering that Euclidean distance $d(\mathbf{z}_1, \mathbf{z}_2)$ is used in all of them, whose complexity is $3d$ if \mathbf{z}_1 and \mathbf{z}_2 are d -dimensional, then we have:

1. **k-NN**: distances to all prototypes have to be calculated ($2nd$) and the minimum will have to be stored in a sorted list of k nearest prototypes ($n \cdot \log_2 k$). The total complexity therefore is $n(3d + \log_2 k)$.
2. **k-NFL**: projection points onto lines (Eq. (3)) have to be calculated ($14dn_L$) and also distances to all feature lines ($2dn_L$). The minimum will have to be stored in a sorted list of k nearest prototypes ($n_L \cdot \log_2 k$). In consequence, the total complexity is $n_L(16d + \log_2 k)$.
3. **k-NFP**: projection points on planes (Eq. (5)) have to be calculated ($n_F O(30d+36)$) and also distances to all feature planes ($2dn_F$). The minimum will have to be stored in a sorted list of k nearest prototypes ($n_F \cdot \log_2 k$). The total complexity therefore is $n_F(2d + \log_2 k + O(30d+36))$.
4. **NFS**: projection points on spaces (Eq. (7)) have to be calculated ($C(4n_c d + n_c^2(2d+1) + n_c^3)$) and also distances to all feature spaces ($2dC$) and the minimum will have to be found ($C \cdot \log_2 C$). Consequently, the total complexity is $C(4n_c d + n_c^2(2d+1) + n_c^3 + 2d + \log_2 C)$.

3.2 Error-Complexity Curves

Complexity can be empirically studied by exploring the error-complexity trade-off. As in a cost-benefit analysis, a series of experiments should be conducted, varying the number of prototypes in order to investigate the dependency of the performance on it. For feature extraction, the eigenface representation was applied (cf. Section 1). Obviously, computational complexity can be lowered by retaining as fewer eigenfaces as possible; nonetheless, 40 eigenfaces are sufficient for a very good description of the training set (Chin & Suter, 2004).

We have used $k=1$ for all the classifiers; nonetheless, k could be optimized by the leave-one-out procedure. Data sets used here as examples are:

- The AT&T (previously ORL) Database of Faces, with $C=40$, $d=93 \times 112$ pixels which was reduced up to $d=40$ and $n \in C \cdot \{3,4,5\}$.
- The Sheffield (previously UMIST) Face Database, with $C=20$, $d=93 \times 112$ pixels which was reduced up to $d=40$ and $n \in C \cdot \{3,5,7,9\}$.

Error e (in %, measured on an independent set of 5 examples per class) vs. computational complexity f (in FLOPs) for the AT&T database of faces is shown in Figure 3. Similarly, the error e (in %, measured on an independent set of 9 examples per class) vs. computational complexity f (in FLOPs) for the Sheffield Face Database is shown in Figure 4. In both cases the number of FLOPs corresponds to the classification of a single example z .

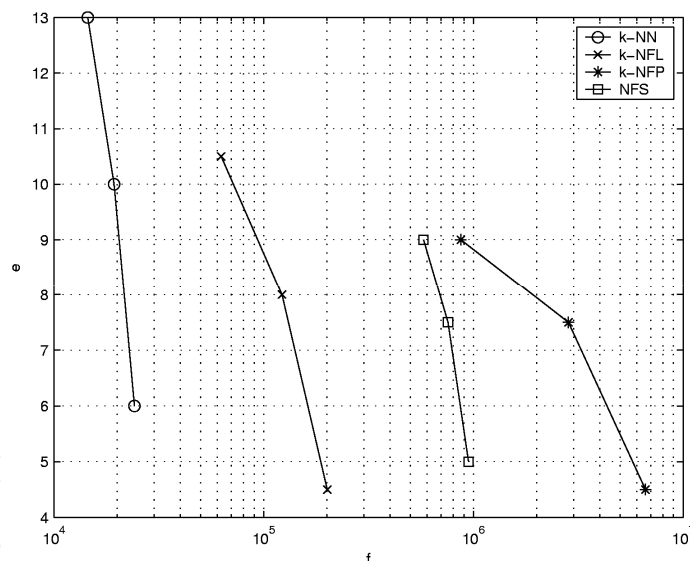


Figure 3. Classifier complexity f (in FLOPs) vs. e (in %) for the AT&T database of faces

When NFS is applied for three prototypes per class, it becomes k-NFP. It is noteworthy that performance could decline. In practice, k-NFP classifier is not advisable because its computational complexity becomes too high. If a sufficient number of prototypes is available, for example 5 or 9 prototypes per class for each database, the best choice would be k-NFL. Only in those cases where the number of prototypes is not large enough to cover variations for each object, the more expensive nearest feature classifiers should be used.

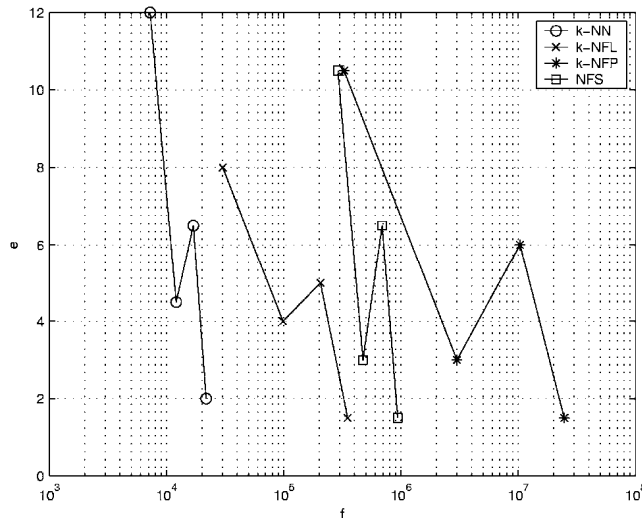


Figure 4. Classifier complexity f (in FLOPs) vs. e (in %) for the AT&T database of faces

3.3 The economics of nearest feature classification

In (de Ridder et al., 2002), a simple economic model for comparing classification error to computational complexity was proposed. According to such approach, cost of classification errors (ϵ_c) and cost of complexity (ϵ_p) can be compared by the cost of a single error (ϵ_0) as follows:

$$c_e = c_p, \quad (28)$$

$$e \cdot c_0 = \frac{c_c \cdot f}{3.15 \times 10^7 \cdot v \cdot s} = 1.27 \times 10^{-10} \cdot f, \quad (29)$$

where e is the probability of misclassification, v is the number of FLOPs per second (there are 3.15×10^7 seconds in a year), c_c is the total annual cost of ownership for a computer, f is the number of FLOPs needed to classify a single sample z and s is the percentage of CPU time allotted to classification. It was found experimentally that $c_c = \text{€}10^4$, $v = 10^7$ and $s = 0.25$ are reasonable values.

A direct comparison between two classifiers A and B can be done by a slight modification of (29):

$$c_{B,A} = 1.27 \times 10^{-10} \cdot \frac{f_B - f_A}{|e_B - e_A|}, \quad (30)$$

Eq. (30) represents the cost of using classifier B instead of another classifier A. Interesting cases are those with $(e_B - e_A) < 0$; that is, selecting a classifier which improves the performance. For these cases, $c_{B,A} > 0$ indicates that classifier B complexity is larger than that of classifier A and, in consequence, the improvement must be paid. Conversely, if $c_{B,A} < 0$ we would have a cheaper and better classifier. Costs of interesting cases for the AT&T Database of Faces and the Sheffield Face Database are shown in Tables 1 and 2, respectively.

Classifier	A			
	k-NN	k-NFL	k-NFP	NFS
B	3 prototypes per class			
k-NN	—	—	—	—
k-NFL	0.0002	—	—	—
k-NFP	0.0027	0.0068	—	*
NFS	0.0018	0.0044	*	—
	4 prototypes per class			
k-NN	—	—	—	—
k-NFL	0.0007	—	—	—
k-NFP	0.0143	0.0690	—	*
NFS	0.0037	0.0161	*	—
	5 prototypes per class			
k-NN	—	—	—	—
k-NFL	0.0015	—	*	-0.0190
k-NFP	0.0558	*	—	0.1438
NFS	0.0117	—	—	—

*: $e_B = e_A$; —: $(e_B - e_A) > 0$

Table 1. Economics of nearest feature classifiers. AT&T database of faces

Classifier	A			
	k-NN	k-NFL	k-NFP	NFS
B	3 prototypes per class			
k-NN	—	—	—	—
k-NFL	0.0001	—	-0.0015	-0.0013
k-NFP	0.0027	—	—	*
NFS	0.0024	—	*	—
	5 prototypes per class			
k-NN	—	—	—	—
k-NFL	0.0022	—	—	—
k-NFP	0.0253	0.0368	—	*
NFS	0.0039	0.0048	*	—
	7 prototypes per class			
k-NN	—	—	—	*
k-NFL	0.0016	—	-0.1287	-0.0041
k-NFP	0.2623	—	—	0.2452
NFS	*	—	—	—
	9 prototypes per class			
k-NN	—	—	—	—
k-NFL	0.0084	—	*	*
k-NFP	0.6261	*	—	*
NFS	0.0233	*	*	—

*: $e_B = e_A$; —: $(e_B - e_A) > 0$

Table 2. Economics of nearest feature classifiers. Sheffield face database

k-NFP classifier is always the most expensive option. The cheaper solution which gives an acceptable error in comparison with the best possible performance is k-NFL; in fact, in several cases using k-NFL instead of other nearest feature classifier is a saving on the cost of error-complexity. In general, costs of preferring NFS are acceptable.

4. Dissimilarity-based face recognition

The concept of proximity is essential in learning processes. Identifying differences or, conversely, detecting shared commonalities are typically carried out by using a suitable proximity measure, often referred to as a *dissimilarity*. Such a proximity can be modelled in different ways, according to the nature of data; e.g. as a classical distance between vector representations or by using edit distances between structural descriptions, such as shapes or sequences.

A wide-scope approach, the dissimilarity representation for pattern recognition (Pełkalska & Duin, 2005a), was proposed on the basis of such proximity measures. Statistical and structural learning techniques can be directly used with dissimilarity representations, naturally fitting for a variety of applications, e.g. face recognition problems. In addition, since dissimilarity measures are considered very general, they are not constrained to Euclidean or metric behaviors, neither to positive semidefinite structures as it is imposed beforehand in kernel methods. The aim of this Section is to review the practical foundations of the dissimilarity-based approach and to explore its application for a simple face recognition problem.

4.1 Dissimilarity representations

A dissimilarity representation of objects is based on their pairwise comparisons. Consider a representation set $R:=\{p_1,p_2,\dots,p_n\}$ and a dissimilarity measure d . An object x is represented as a vector of the dissimilarities computed between x and the prototypes from R , i.e. $D(x,R)=[d(x,p_1),d(x,p_2),\dots,d(x,p_n)]$. For a set T of N objects, it extends to an $N\times n$ dissimilarity matrix (Pełkalska et al., 2006):

$$D(T,R) = \begin{matrix} & p_1 & p_2 & p_3 & \cdots & p_n \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{matrix} & \begin{pmatrix} d_{11} & d_{12} & d_{13} & \cdots & d_{1n} \\ d_{21} & d_{22} & d_{23} & \cdots & d_{2n} \\ d_{31} & d_{32} & d_{33} & \cdots & d_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & d_{N3} & \cdots & d_{Nn} \end{pmatrix} \end{matrix} \quad (31)$$

where $d_{jk}=D(x_j,p_k)$.

For dissimilarities, the geometry is contained in the definition, giving the possibility to include physical background knowledge; in contrast, feature-based representations usually suppose a Euclidean geometry. Important properties of dissimilarity matrices, such as metric nature, tests for Euclidean behavior, transformations and corrections of non-Euclidean dissimilarities and embeddings, are discussed in (Pełkalska & Duin, 2005b).

When the entire T is used as R , the dissimilarity representation is expressed as an $N\times N$ dissimilarity matrix $D(T,T)$. Nonetheless, R may be properly chosen by prototype selection procedures (Pełkalska et al., 2006).

4.2 Classifiers in dissimilarity spaces

Building a classifier in a dissimilarity space consists in applying a traditional classification rule, considering dissimilarities as features; that is, in practice, a dissimilarity-based classification problem is addressed as a traditional feature-based one. Even though the nearest neighbor rule is the reference method to discriminate between objects represented by dissimilarities, it suffers from a number of limitations. Previous studies (Pękalska et al., 2001; Pękalska & Duin, 2002; Paclík & Duin, 2003; Pękalska et al., 2004; Orozco-Alzate et al., 2006) have shown that Bayesian (normal density based) classifiers, particularly the linear (LDC) and quadratic (QDC) normal based classifiers, perform well in dissimilarity spaces and, sometimes, offer a more accurate solution. For a 2-class problem, the LDC based on the representation set R is given by

$$f(D(x,R)) = \left[D(x,R) - \frac{1}{2}(\mathbf{m}_{(1)} + \mathbf{m}_{(2)}) \right]^T \times C^{-1}(\mathbf{m}_{(1)} - \mathbf{m}_{(2)}) + \log \frac{P_{(1)}}{P_{(2)}} \quad (32)$$

and the QDC is derived as

$$f(D(x,R)) = \sum_{i=1}^2 (-1)^i (D(x,R) - \mathbf{m}_{(i)})^T \times C_{(i)}^{-1} (D(x,R) - \mathbf{m}_{(i)}) + 2 \log \frac{P_{(1)}}{P_{(2)}} + \log \frac{|C_{(1)}|}{|C_{(2)}|} \quad (33)$$

where C is the sample covariance matrix, $C_{(1)}$ and $C_{(2)}$ are the estimated class covariance matrices, and $\mathbf{m}_{(1)}$ and $\mathbf{m}_{(2)}$ are the mean vectors, computed in the dissimilarity space $D(T,R)$. $P_{(1)}$ and $P_{(2)}$ are the class prior probabilities. If C is singular, a regularized version must be used. In practice, the following regularization is suggested for $\lambda=0.01$ (Pękalska et al., 2006):

$$C_{\text{reg}}^\lambda = (1-\lambda)C + \lambda \text{diag}(C) \quad (34)$$

Nonetheless, regularization parameter should be optimized in order to obtain the best possible results for the normal density based classifiers.

Other classifiers can be used in dissimilarity spaces, usually by a straightforward implementation. Nearest mean linear classifiers, Fisher linear discriminants, support vector machines (SVMs), among others are particularly interesting for being used in generalized dissimilarity spaces. In addition, traditional as well as specially derived clustering techniques can be implemented for dissimilarity representations, see (Pękalska & Duin, 2005c) for a detailed discussion.

4.3 Experimental results

As in Section 3.2, experiments were conducted on the AT&T and the Sheffield datasets, using 40 eigenfaces for an initial representation. Dissimilarity representations were constructed by calculating pairwise Euclidean distances on the eigenface representations. In order to compare different classifiers, the k-NN rule and the LDC and QDC classifiers built on the dissimilarity representations were used. Experiments were performed 25 times for randomly chosen training and test sets. Since in this study we are particularly interested in recognition accuracy rather than in computational complexity and storage requirements, the entire training set T has been used as the representation set R . Nonetheless, R may be properly reduced by prototype selection procedures (Pękalska et al., 2006). Training and testing sets were generated by selecting equal partitions for the classes.

Figures 5 and 6 present the results, in terms of classification errors as a function of the number of training objects randomly chosen. Figure 5 presents the results for the AT&T database; similarly, the results for the Sheffield dataset are shown in Figure 6. Standard deviations for averaged test error decrease rapidly, varying around 0.15 and 0.08 after at least 6 training objects per class are available; for clarity reasons, standard deviations are not given.

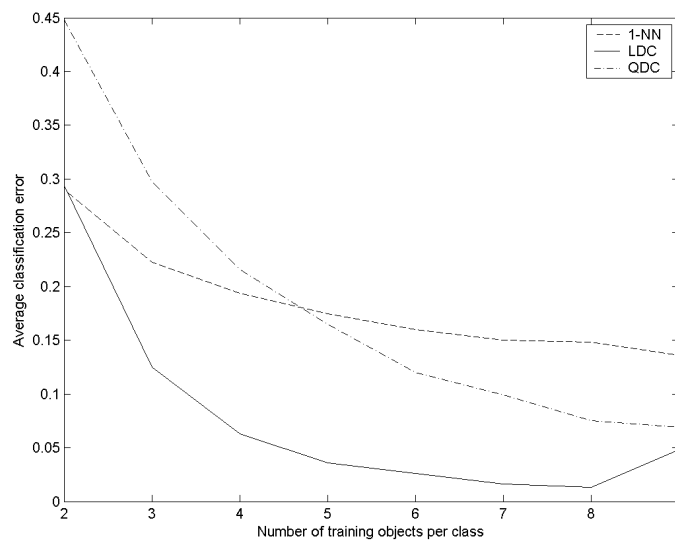


Figure 5. Average classification error as a function of the number of prototypes per class for the ORL database of faces

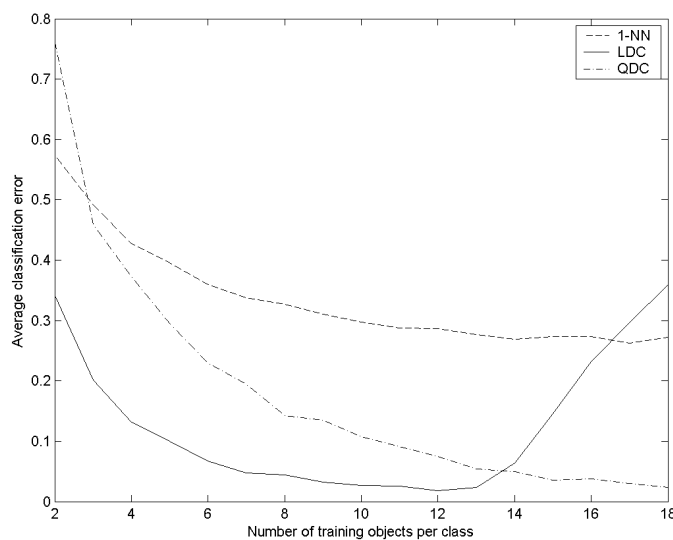


Figure 6. Average classification error as a function of the number of prototypes per class for the Sheffield database of faces

Experiments confirm that Bayesian classifiers outperform the 1-NN classifier, whenever a sufficient number of prototypes is available. Moreover, LDC for both data sets outperforms the 1-NN rule and the QDC; nonetheless, it shows a loss of accuracy when certain number of prototypes is provided. Therefore, a further study on a proper regularization for the LDC should be conducted.

5. Nearest feature rules and dissimilarity representations

Recently, a number of research advances on dissimilarity representations has been carried out. They showed that learning from dissimilarity representations is a feasible alternative to learning from feature-based descriptions (Pełkalska & Duin, 2002; Paclík & Duin, 2003; Pełkalska & Duin, 2005a). In spite of those remarkable advances, the work is not completed yet; particularly, meaningful transformations and manipulations of dissimilarity representations are still an open and promising field for future research. Particularly, manipulations to enrich the original dissimilarity representations might be useful; e.g. by using a geometrical generalization.

In such a way, a dissimilarity representation of an object x , which is defined as a set of dissimilarities between x and the objects of a collection $R:=\{p_1, p_2, \dots, p_n\}$, expressed as a vector $D(x, R)=[d(x, p_1), d(x, p_2), \dots, d(x, p_n)]$, is generalized by considering a new set R composed by objects lying in another space, e.g. lines or planes. Considering such a generalized representation, the entire scope of pattern recognition can be studied: representation, data understanding, transformations, classification, etc. In addition, new applications should be considered in order to describe other pattern recognition problems where dissimilarity representations and generalized dissimilarity representations might be advantageous. In summary, the task consists in studying classification in generalized dissimilarity representations; that is, constructing classifiers on spaces equipped with a dissimilarity measure $\rho: X \times X_g \rightarrow \mathbf{R}$, where X_g stands for a generalization of X . In general, dimension of X_g is higher than that of X .

5.1 Generalization of Dissimilarity Representations

The generalization consists in creating matrices $D_L(T, R_L)$ and $D_F(T, R_F)$ by using the information available at the original representation $D(T, R)$. $D_L(T, R_L)$ and $D_F(T, R_F)$ are called *generalized dissimilarity representations* and their structures are:

$$D_L(T, R_L) = \begin{matrix} & \begin{matrix} L_1 & L_2 & L_3 & \dots & L_n \end{matrix} \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{matrix} & \begin{pmatrix} d_{11} & d_{12} & d_{13} & \dots & d_{1n_L} \\ d_{21} & d_{22} & d_{23} & \dots & d_{2n_L} \\ d_{31} & d_{32} & d_{33} & \dots & d_{3n_L} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & d_{N3} & \dots & d_{Nn_L} \end{pmatrix} \end{matrix} \quad (35)$$

where $d_{jk}=D_L(x_j, L_k)$; and

$$D_F(T, R_F) = \begin{matrix} & F_1 & F_2 & F_3 & \dots & F_n \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{matrix} & \begin{pmatrix} d_{11} & d_{12} & d_{13} & \dots & d_{1n_F} \\ d_{21} & d_{22} & d_{23} & \dots & d_{2n_F} \\ d_{31} & d_{32} & d_{33} & \dots & d_{3n_F} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & d_{N3} & \dots & d_{Nn_F} \end{pmatrix} \end{matrix} \quad (36)$$

where $d_{jk} = D_F(x_j, F_k)$.

$D(T, R_L)$ and $D(T, R_F)$ are high dimensional matrices because the original representation set R is generalized by combining all the pairs (R_L) and all the triplets (R_F) of prototypes of the same class. In consequence, a suitable procedure for feature selection (dimensionality reduction) is needed in order to avoid the curse of the dimensionality.

A dissimilarity matrix $D(T, R) = (d_{ij})$ is composed of $C \times C$ submatrices as follows:

$$D(T, R) = \begin{pmatrix} D^{11} & D^{12} & \dots & D^{1C} \\ D^{21} & D^{22} & \dots & D^{2C} \\ \vdots & \vdots & \ddots & \vdots \\ D^{C1} & D^{C2} & \dots & D^{CC} \end{pmatrix}, \quad (37)$$

where D^{ii} and D^{ij} , $i \neq j$ contain intraclass and interclass distances respectively. All the possible dissimilarities between objects are available but the original feature points are not. Nonetheless, it is possible to compute the distances to feature lines from the dissimilarities. The problem consists in computing the height of a scalene triangle as shown in Figure 7.

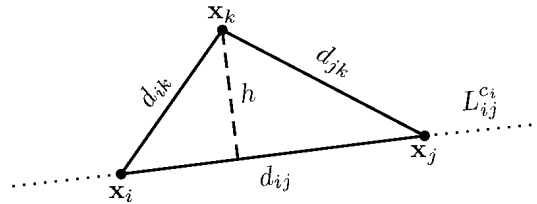


Figure 7. Height of a scalene triangle corresponding to the distance to a feature line

Let us define $s = (d_{jk} + d_{ij} + d_{ik})/2$. Then, the area of the triangle is given by:

$$A = \sqrt{s(s - d_{jk})(s - d_{ij})(s - d_{ik})}; \quad (38)$$

but we also know that area, assuming d_{ij} as base, is:

$$A = \frac{d_{ij} h}{2} \quad (39)$$

So, we can solve (38) and (39) for h , which is the distance to the feature line. The generalized dissimilarity representation in (35) is constructed by replacing each entry of $D(T, R_L)$ by the

corresponding value of h . The distance d_{ij} in Figure 7 must be an intraclass one; that is, $d_{ij} \in D^i$.

Computing the distances to the feature planes in terms of dissimilarities consists in calculating the height of an irregular (scalene) tetrahedron as shown in Figure 8.

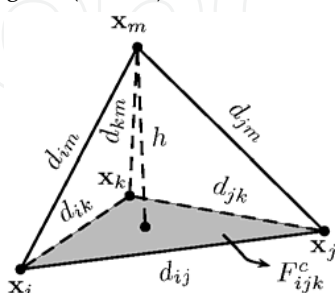


Figure 8. Height of an irregular tetrahedron corresponding to the distance to a feature plane

Let us define $s = (d_{jk} + d_{ij} + d_{ik})/2$. Then, the volume of a tetrahedron is given by:

$$V = \frac{h \sqrt{s(s-d_{jk})(s-d_{ij})(s-d_{ik})}}{3} \quad (40)$$

but volume is also (Uspensky, 1948):

$$V^2 = \frac{1}{288} \begin{vmatrix} 0 & d_{ij}^2 & d_{ik}^2 & d_{im}^2 & 1 \\ d_{ij}^2 & 0 & d_{jk}^2 & d_{jm}^2 & 1 \\ d_{ik}^2 & d_{jk}^2 & 0 & d_{km}^2 & 1 \\ d_{im}^2 & d_{jm}^2 & d_{km}^2 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{vmatrix} \quad (41)$$

So, we can solve (40) and (41) for h , which is the distance to the feature plane. The generalized dissimilarity representation in (36) is constructed by replacing each entry of $D(T, R_F)$ by the corresponding value of h . Distances d_{ij} , d_{ik} and d_{jk} in Figure 8 must be intraclass.

Experiments have shown that nearest feature rules are especially profitable when variations and conditions are not fully represented by the original prototypes; for example the case of small or non-representative training sets. The improvement in such a case respect to the k-NN rule (the reference method) is due to the feature lines/planes' ability to expand the representational capacity of the available points, accounting for new conditions not represented by the original set (Li & Lu, 1999; Chien and Wu, 2002; Orozco-Alzate, 2005; Orozco-Alzate & Castellanos-Domínguez, 2006). Those are precisely the conditions in face recognition problems, where the number of prototypes is typically limited to few images per class and the number of classes is high: tens or even one hundred people. As a result, the effectiveness of the nearest feature rules is remarkable for this problem.

Representations to be studied include generalizations by feature lines, feature planes and the feature space. These representations are not square, having two or three zeros per column for feature lines and feature planes respectively. First, generalizations of metric

representations will be considered because the generalization procedure requires constructing triangles and tetrahedrons and, as a consequence, generalizing non-metric dissimilarity representations might produce complex numbers when solving equations for heights.

To construct classifiers based on generalized dissimilarity representations, we should proceed similarly as dissimilarity-based classifiers are built. That is, using a training set T and a representation set R containing prototype examples from T . Prototype lines or planes considered will be selected by some prototype selection procedure; classifiers should be built on $D(T, R_L)$ and $D(T, R_F)$. Different sizes for the representation set R must be considered. Enriching the dissimilarity representations implies a considerable number of calculations. The number of feature lines and planes grows rapidly as the number of prototypes per class increases; in consequence, computational effort may become high, especially if a generalized representation is computed for an entire set. When applying traditional statistical classifiers to dissimilarity representations, dissimilarities to prototypes may be treated as features. As a result, classifiers built in enriched dissimilarity spaces are also subject to the curse of dimensionality phenomenon. In general, for generalized dissimilarity representations $D_g(T, R_g)$, the number of training objects is small relative to the number of prototype lines or planes.

According to the two reasons above, it is important to use dimensionality reduction techniques – feature extraction and feature selection methods – before building classifiers in generalized dissimilarity representations. Systematic approaches for prototype selection such as exhaustive search and the forward selection process lead to an optimal representation set; however, they require a considerable number of calculations. Consequently, due to the increased dimensionality of the enriched representations, the application of a systematic prototype selection method will be computationally expensive. Nonetheless, it has been shown that non-optimal and computationally simple procedures such as *Random* and *RandomC* may work well (Pękalska et al., 2006).

6. Conclusion

In this chapter, we presented a series of theoretical and experimental considerations regarding the nearest feature rules and dissimilarity representations for face recognition problems, analyzed separately as well as a combined approach. Firstly, a study about the asymptotic behavior of the nearest feature classifiers was conducted, following the well-known procedure derived for the k -NN rule. We concluded that, if an arbitrarily large number of samples is available, there is no significant difference between k -NN and its geometric generalizations: the nearest feature rules. Moreover, as for k -NN, it is not possible to say something general about the asymptotic behavior in the finite-sample case. It might be possible to perform an analysis for specific distributions; perhaps without loss of generality. Consequently, further conceptual considerations and experiments are required.

Quantifying the computational complexity of classifiers is very important in the selection of a particular algorithm. Complexity of algorithms is usually measured in terms of orders; nonetheless, such an approach is not precise. An evaluation of the error-complexity trade-off for the nearest feature classifiers has been presented in Section 3. We have also studied the complexity of nearest feature classifiers, in terms of the number of additions and multiplications associated to their evaluation, as well as through error-complexity curves and a comparative study considering error and complexity. It was shown that k -NFP is too

expensive for practical applications and that k-NFL and NFS are better options to overcome the representational limitations of k-NN. Even though nearest feature rules are well-performing classifiers, their computational complexity is too high. If there is a maximum acceptable response delay for the particular application and a considerable number of prototypes is available, an effective way to overcome this shortcoming might be to use parallel computation.

We have explored and tested a dissimilarity-based strategy for face recognition. Two simple classification problems were conducted: the classic ORL database and the Sheffield data set. Dissimilarity representation was derived by applying the eigenface transformation and, afterwards, the Euclidean distance between the eigenface representations. Such a representation allowed us for using traditional statistical decision rules, particularly normal density based classifiers. The 1-NN rule was employed as a reference for performance comparison. Those experiments confirm that Bayesian classifiers outperform the 1-NN classifier, when a sufficient number of prototypes is provided. The LDC constructed for both the ORL and the Sheffield problems, always outperforms the 1-NN rule; however, LDC shows a loss of accuracy when certain number of prototypes is provided. Therefore, a further study on a proper regularization for the LDC should be conducted in order to obtain an improvement of this classifier.

Finally, an approach to combine the nearest feature rules and dissimilarity representations was proposed. There are several ways to use the nearest feature rules for enriching a given dissimilarity representation. To begin with, we suggested considering generalizations by feature lines and feature planes, restricted to metric dissimilarities in order to avoid complex numbers when solving equations for heights. However, such a restriction can be overcome by using Euclidean embeddings. In addition, combined classifiers seem to be an option because a new and extended representation can be constructed by combining the original and the generalized ones. As a result, there are several fundamental and applied research problems to be faced in future research work.

7. References

- Chien, J.-T. and Wu, C.-C. (2002). Discriminant waveletfaces and nearest feature classifiers for face recognition. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(12):1644-1649.
- Chin, T. and Suter, D. (2004). A study of the eigenface approach for face recognition. Technical report, Monash University.
- Cover, T. M. and Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Trans. Inform. Theory*, IT-13(1):21-27.
- de Ridder, D., Pękalska, E., and Duin, R. (2002). The economics of classification: Error vs. complexity. In *Proceedings of the 16th International Conference on Pattern Recognition*, volume 2, pages 244-247, Quebec, Canada.
- Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification*. Wiley-Interscience, 2 edition.
- Hart, P. E. (1968). The condensed nearest neighbor rule. *IEEE Transactions on Information Theory*, IT-14(3):515-516.
- Li, S. Z. and Lu, J. (1999). Face recognition using the nearest feature line method. *IEEE Trans. Neural Networks*, 10(2):439-443.

- Orozco-Alzate, M. (2005). Selección de características wavelet para clasificación de señales 1-D y 2-D usando algoritmos genéticos. *Master's thesis*, Universidad Nacional de Colombia Sede Manizales.
- Orozco-Alzate, M. and Castellanos-Domínguez, C. G. (2006). Comparison of the nearest feature classifiers for face recognition. *Machine Vision and Applications*, 17(5):279–285.
- Orozco-Alzate, M., García-Ocampo, M. E., Duin, R. P. W., and Castellanos-Domínguez, C. G. (2006). Dissimilarity-based classification of seismic volcanic signals at Nevado del Ruiz volcano. *Earth Sciences Research Journal*, 10(2):57-65.
- Paclík, P. and Duin, R. P. W. (2003). Dissimilarity-based classification of spectra: computational issues. *Real Time Imaging*, 9:237–244.
- Pękalska, E. and Duin, R. P. W. (2002). Dissimilarity representations allow for building good classifiers. *Pattern Recognition Lett.*, 23:943–956.
- Pękalska, E. and Duin, R. P. W. (2005a). *The Dissimilarity Representation for Pattern Recognition: Foundations and Applications*. World Scientific, Singapore.
- Pękalska, E. and Duin, R. P. W. (2005b). *The Dissimilarity Representation for Pattern Recognition: Foundations and Applications*, chapter 3: Characterization of dissimilarities, pages 89–145. World Scientific.
- Pękalska, E. and Duin, R. P. W. (2005c). *The Dissimilarity Representation for Pattern Recognition: Foundations and Applications*, chapter 7: Further data exploration, pages 289–332. World Scientific.
- Pękalska, E., Duin, R. P. W., Günter, S., and Bunke, H. (2004). On not making dissimilarities Euclidean. In *Proceedings of Structural and Statistical Pattern Recognition*, pages 1143–1151, Lisbon, Portugal.
- Pękalska, E., Duin, R. P. W., and Paclík, P. (2006). Prototype selection for dissimilarity-based classifiers. *Pattern Recognition*, 39:189–208.
- Pękalska, E., Paclík, P., and Duin, R. P. W. (2001). A generalized kernel approach to dissimilarity based classification. *J. Mach. Learn. Res.*, 5:175–211.
- Uspensky, J. V. (1948). *Theory of Equations*. McGraw-Hill.



Face Recognition

Edited by Kresimir Delac and Mislav Grgic

ISBN 978-3-902613-03-5

Hard cover, 558 pages

Publisher I-Tech Education and Publishing

Published online 01, July, 2007

Published in print edition July, 2007

This book will serve as a handbook for students, researchers and practitioners in the area of automatic (computer) face recognition and inspire some future research ideas by identifying potential research directions. The book consists of 28 chapters, each focusing on a certain aspect of the problem. Within every chapter the reader will be given an overview of background information on the subject at hand and in many cases a description of the authors' original proposed solution. The chapters in this book are sorted alphabetically, according to the first author's surname. They should give the reader a general idea where the current research efforts are heading, both within the face recognition area itself and in interdisciplinary approaches.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Mauricio Orozco Alzate and German Castellanos Dominguez (2007). Nearest Feature Rules and Dissimilarity Representations for Face Recognition Problems, *Face Recognition*, Kresimir Delac and Mislav Grgic (Ed.), ISBN: 978-3-902613-03-5, InTech, Available from:

http://www.intechopen.com/books/face_recognition/nearest_feature_rules_and_dissimilarity_representations_f_or_face_recognition_problems

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2007 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen