On Selecting Middle-Length Feature Lines for Dissimilarity-based Classification

Mauricio Orozco-Alzate, Robert P. W. Duin, and César Germán Castellanos-Domínguez

***Abstract*—Raw or preprocessed measurements, such as signals and images, must be properly represented before using computer methods for learning or classification. Feature-based representations are traditionally used. An alternative is to build a dissimilarity representation; that is, to describe the objects in terms of measures of pairwise comparisons, which are referred to a set of representative objects called prototypes. Given a dissimilarity representation computed from a very small set of prototypes, an option to overcome representational limitations is the use of feature lines resulting from the the linear combination of pairs of prototypes. The choice of a proper subset of feature lines is an important issue, not just to obtain a good description but also to reduce the dimensionality. In this paper, we consider the selection of the middle-length feature lines, comparing the results to those obtained when the longest lines are selected. A number of experiments has been conducted on various artificial and real-world data sets. In general, we find out that the middle-length feature lines are more appropriate to represent moderately curved subspaces.**

***Key Words*—Classification, dissimilarity representation, feature line, pattern recognition, selection.**

I. Introduction

HOW to learn from sensor measurements of a few examples of objects, e.g. signals or images belonging to a number of classes, is the main interest in the study of automatic pattern recognition. A crucial issue in this discipline is to derive an appropiate mathematical representation from the measurements. Two different but related approaches for obtaining such a representation can be considered: the traditional way based on numerical features for each particular object and the alternative one of representing objects in terms of their dissimilarities to a set of prototypes. In the first approach, objects are represented as points in a feature vector space; in the second one, each dimension of the vectors corresponds to a dissimilarity measure resulting from a pairwise comparison.

The nearest neighbor rule (1-NN) [1] is the classification procedure typically applied to dissimilarities. In spite of its simplicity and good asymptotic behavior, the applicability is restricted under representational limitations, presence of noise and demanding specifications such as storage and computational effort. An alternative approach to learn from dissimilarities —the so-called dissimilarity representations (DRs)— was recently proposed [2], [3]. Such an approach basically consists in using the dissimilarities to define a space and, afterwards, constructing classifiers directly on it; for instance, normal density based classifiers. One of the advantageous properties of the DRs is the possibility to exploit larger training sets, increasing the accuracy while the complexity remains the same. A different approach to overcome the limitations associated to the 1-NN rule is the also recent development of the nearest feature classifiers [4]–[6]. Such classifiers are geometric extensions of the 1-NN rule. The nearest feature classifiers, in their basic setup, encompass the nearest feature line (NFL) and the nearest feature plane (NFP) classifiers, which aim at enriching the representation through the interpolation and extrapolation between pairs and triples of feature points.

In a previous study [7], we propose to combine both strategies, namely DRs and NFL, in order to take advantage of their individual benefits. The combined approach leads to the so-called generalized dissimilarity representations (GDRs) by feature lines, which in brief consists in using feature lines as prototypes instead of feature points and then to build a classifier on that representation. Since the number of feature lines grows combinatorially, a strong regularization for the classifiers must be used in order to counteract the effect of the peaking phenomenon. In this paper, we study an alternative procedure to deal with the dimensionality problem associated to the feature lines. We propose to rank the feature lines according to their length and then, to select those having middle lengths; that is, extracting a subset of feature lines placed in the middle of the ranking. In comparison with the selection of largest and/or shortest feature lines as we explore previously, the middle-length ones seem to be more suitable to get a piecewise description of curved subspaces. Our observations and discussions are supported by a series of experiments with elongated or correlated data sets, which are the type of problems naturally benefited by the generalization using feature lines.

M. Orozco-Alzate and C. G. Castellanos-Domínguez are with the Control and Digital Signal Processing Group, Faculty of Engineering and Architecture, National University of Colombia at Manizales, Kilómetro 7 Vía al Aeropuerto, Campus La Nubia - Bloque Q, piso 2, Manizales (Caldas), Colombia. See `http://www.docentes.unal.edu.co/morozcoa/` or `http://orozco.co.nr/` for contact details.

Robert P. W. Duin is with the Information and Communication Theory (ICT) group in the Faculty of Electrical Engineering, Mathematics and Computer Science. Delft University of Technology, The Netherlands. See `http://www-ict.ewi.tudelft.nl/~duin/` for contact details.

II. Generalization of Dissimilarity Representations Using Feature Lines

In this Section, we describe our procedure for generalizing dissimilarity representations. Before explaining the algorithm itself, the 1-NN and NFL are reviewed as well as the alternative approach of learning from dissimilarity representations.

A. The 1-NN and NFL Rules

Before defining the 1-NN and NFL rules, consider the following brief comment on notation. The usual way to denote the set of class labels is $\Omega = \{\omega_1, \ldots, \omega_c\}$. However, for the sake of simplicity, we denote the membership or association to one of the C classes by using the letter c, as a variable running from 1 to C. In addition, when a particular value of c is used as a subscript, it is written within parentheses. Taking into account such a notation, we define 1-NN as a rule that classifies an object x by assigning it the class label $\hat{c}$ associated to the nearest training object. In a feature space representation, x is represented as a feature vector $\boldsymbol{x}$. Considering a training set $T = \{\boldsymbol{x}_i^c, 1 \leq c \leq C, 1 \leq i \leq n_c\}$, where C is the number of classes and n_c the number of objects per class, the rule can then be written as follows:

$$d(\boldsymbol{x}, \boldsymbol{x}_{\hat{i}}^{\hat{c}}) = \min_{1 \leq c \leq C,\ 1 \leq i \leq n_c} d(\boldsymbol{x}, \boldsymbol{x}_i^c), \tag{1}$$

where $d(\boldsymbol{x}, \boldsymbol{x}_i^c) = \|\boldsymbol{x} - \boldsymbol{x}_i^c\|$ is usually the (weighted) Euclidean or the city block norm. Several variations have been proposed to enhance the 1-NN rule, e.g. the editing and condensing rules (see [8] for a comprehensive review and comparison of these techniques) and the nearest feature classifiers [4], [5]. From these last variations, the NFL classifiers has received a considerable attention in the pattern recognition field, showing its good performance in many applications such as face recognition, audio retrieval, image classification, speaker identification and object recognition [9]. In this study, we particularly focus on that classfier, as an intermediate tool to generalize dissimilarity representations.

The NFL classifier [4], is an extension of the 1-NN method. It generalizes each pair of prototype feature points belonging to the same class: $\{\boldsymbol{x}_i^c, \boldsymbol{x}_j^c\}$ by a linear function L_{ij}^c, which is called the *feature line*. Such a line covers the subspace spanned by the pair of points; that is, $L_{ij}^c = \mathrm{sp}(\boldsymbol{x}_i^c, \boldsymbol{x}_j^c)$. In order to classify a query $\boldsymbol{x}$, it is projected onto L_{ij}^c as a point $\tilde{\boldsymbol{x}}_{ij}^c$ (see Fig. 1). This projection can be computed as

$$\tilde{\boldsymbol{x}}_{ij}^c = \boldsymbol{x}_i^c + \tau(\boldsymbol{x}_j^c - \boldsymbol{x}_i^c), \tag{2}$$

where $\tau = (\boldsymbol{x} - \boldsymbol{x}_i^c) \cdot (\boldsymbol{x}_j^c - \boldsymbol{x}_i^c) / \|\boldsymbol{x}_j^c - \boldsymbol{x}_i^c\|^2 \in \mathbb{R}$; τ is called the *position parameter*.

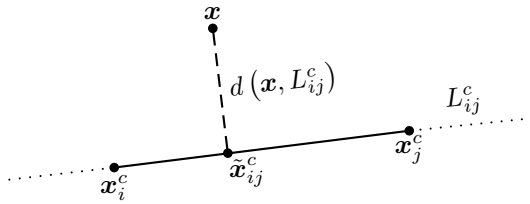


Fig. 1. Computation of the distance to a feature line L_{ij}^c.

Afterwards, the decision is done by assigning to $\boldsymbol{x}$ the class label $\hat{c}$ associated to the nearest feature line; that means:

$$d(\boldsymbol{x}, L_{\hat{i}\hat{j}}^{\hat{c}}) = \min_{\substack{1 \leq c \leq C,\ 1 \leq i,j \leq n_c \\ i \neq j}} d(\boldsymbol{x}, L_{ij}^c) \tag{3}$$

where $d(\boldsymbol{x}, L_{ij}^c) = \|\boldsymbol{x} - \tilde{\boldsymbol{x}}_{ij}^c\|$.

B. Dissimilarity representations

A dissimilarity representation of an object x is a set of dissimilarities between x and the objects of a representation set R, which is composed by n prototypes: $R = \{p_1, p_2, \ldots, p_n\}$ [3]. Under the most general conditions, such dissimilarities may be derived from the objects directly, their measurements, or some intermediate representation; for instance, from an initial feature representation. As a result, the set of n dissimilarities from x to the prototypes constitutes the dissimilarity representation of x. It can be written as a vector $D(x, R) = [d(x, p_1), d(x, p_2), \ldots, d(x, p_n)]$. For a training set T of N objects, it extends to an $N \times n$ dissimilarity matrix $D(T, R)$ [2]. Moreover, the $N \times N$ matrix $D(T, T)$ is a complete representation. Although usually R is a subset of T ($R \subseteq T$), they might be disjunct. Assuming r_c prototypes per class, the cardinality of R is $n = \sum_{c=1}^{C} r_c$.

A dissimilarity representation $D(T, R)$ can be considered also as a data-depending mapping $D(\cdot, R) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^n$ to the so-called *dissimilarity space*, which is defined by R. In such a space, each dimension $D(\cdot, p_i)$ corresponds to a dissimilarity to a particular prototype. Given $D(T, R)$, a test set S of new incoming objects is provided in terms of their dissimilarities related to R, i.e. as a matrix $D(S, R)$. In this approach, the 1-NN rule in the original representation consists in finding the minimum in each row of $D(S, R)$ and assigning to x the class associated to the column where the minimum value is found. Notice that the 1-NN rule is not performed in the dissimilarity space. It is just a more general statement than that given in (1). The issue of building classifiers in the dissimilarity space, taking advantage of the whole information available at T, is discussed below.

C. Classifiers in dissimilarity spaces

Dissimilarities, by definition, should be small for similar objects and large for different ones. In consequence, they provide discriminative information and allow for building classifiers in the dissimilarity space. Any classifier defined in vector spaces can be straightforwardly used in the dissimilarity space. In particular, the use of normal density based classifiers in dissimilarity spaces is suggested because the summation-based distances are often approximately distributed according to a clipped normal distribution [3]. Besides, a linear classifier in a dissimilarity space is equivalent to a non-linear one in the underlying (original) space. In other words, a linear classifier in the dissimilarity space is expected to perform as good as a non-linear one in the original space, while having a computational complexity comparable to that of the 1-NN rule [8].

For a two-class problem, a linear normal density based classifier (BayesNL) based on R is defined by

$$\begin{aligned} f(D(x, R)) = &\left[D(x, R) - \frac{1}{2}\left(\boldsymbol{m}_{(1)} + \boldsymbol{m}_{(2)}\right)\right]^T \\ &\times \boldsymbol{C}^{-1}\left(\boldsymbol{m}_{(1)} - \boldsymbol{m}_{(2)}\right) + \log\frac{P_{(1)}}{P_{(2)}}, \end{aligned} \tag{4}$$

where $\boldsymbol{C}$ is the sample covariance matrix, $\boldsymbol{m}_{(1)}$ and $\boldsymbol{m}_{(2)}$ are the mean vectors and $P_{(1)}$, $P_{(2)}$ are the class prior

probabilities. When the covariance matrices become singular, they must be regularized by using, for example, the following strategy [10]: $\boldsymbol{C}_{reg}^{\lambda} = (1-\lambda)\,\boldsymbol{C} + \lambda \mathrm{diag}(\boldsymbol{C})$. The following suboptimal value is suggested for practical applications: $\lambda \leq 0.01$ [11].

D. Generalization Procedure

Generalizing $D(T,R)$ consists in creating the generalized dissimilarity representation $D_L(T,R_L)$, where the subscript L denotes that R_L is composed by feature lines instead of points. Analogously to the description in Sec. II-B, for a generalized dissimilarity space, the considered mapping is $D(x,R_L) : \mathcal{X} \times \mathcal{X}_L \rightarrow \mathbb{R}^{n_L}$. Therefore, the *generalized dissimilarity representation* of x is the vector $D(x,R_L) = [d(x,L_1), d(x,L_2), \ldots, d(x,L_{n_L})]$, where $n_L = \sum_{c=1}^{C} r_c(r_c-1)/2$.

To be consistent with the general scope of dissimilarity representations, we should not assume that an accompanying feature representation is always available. Thereby, we should derive $D(\cdot,R_L)$ using just the information available at $D(\cdot,R)$ instead of applying (2) and (3). Such a problem can be addressed geometrically as follows: Consider the triangle in Fig. 2, deriving the distances to feature lines consists in computing the height h of such a scalene triangle. Since any metric triplet $\{d_{ij},\ d_{ik},\ d_{jk}\}$ constitutes a Euclidean triangle, we should either restrict our experiments to metric distance matrices or correct them to be Euclidean. Let define $s = (d_{jk}+d_{ij}+d_{ik})/2$. Then, the area of the triangle is given by:

$$A = \sqrt{s(s-d_{jk})(s-d_{ij})(s-d_{ik})}; \tag{5}$$

but it is also known that area, considering d_{ij} as base, is:

$$A = \frac{d_{ij}h}{2} \tag{6}$$

We can solve (5) and (6) for h, which is the distance to the feature line, i.e. $d(x_k, L_{ij}^c)$. The generalized dissimilarity representation for a particular object x_k is constructed by arranging the n_L distances in a vector $D(x_k,R_L)$. For a training set T, we have a $N \times n_L$ generalized dissimilarity matrix $D(T,R_L)$. In general, $D(T,R_L)$ is not square and has two zeros elements per column. The information on a set S of new test objects is provided in terms of their distances to R_L and arranged as a matrix $D(S,R_L)$.

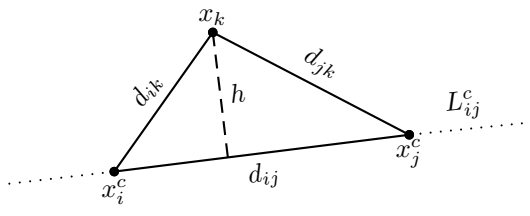


Fig. 2. Scalene triangle for computing the distance to a feature line in terms of dissimilarities.

There is no practical difference between building a linear normal density based classifier in the original dissimilarity space or in the generalized one. The BayesNL definition when applied in the generalized dissimilarity space is the same as that defined in Sec. II-C. It is just needed to replace R by R_L in (4) where appropiate. Finally, notice that $D(T,R_L)$ may be high dimensional due to the combinatorial increase of the number of feature lines. Such a potential problem can be overcome by strongly regularizing the classifier or by selecting, according to some criterion, a subset of feature lines. Here we use both strategies to control the peaking phenomenon; particularly, we use a length-based selection that we proposed in [7], but now selecting the middle-length lines instead of the longest ones as we did in our previous work. A detailed explanation of this selection criterion is given in Sec. III-B.

III. Experiments and Results

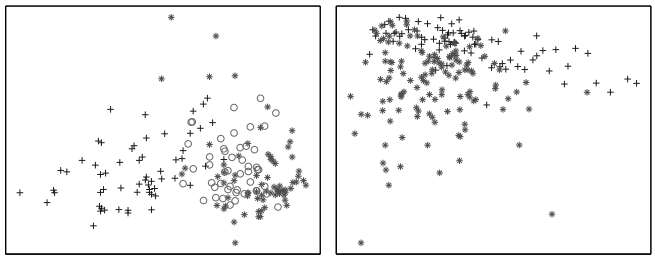
In all our experiments, we derive the initial dissimilarities from the corresponding feature representations, particularly using Euclidean distances in order to meet the metric constraint mentioned above. The reported results shown in Figs. 5–8 are based on 25 repetitions. A strong regularization of $\lambda = 0.01$ was used for all cases. We do not present the resulting standard deviations to keep the plots clear; however, we found that, in general, those deviations vary between 1% and 6% of the averaged errors. Before discussing the results in more detail, the data sets used for the experiments and the length-based selection method are presented.

A. Data Sets

The *Wine* data come from the Machine Learning Repositary [12] and describe three types of wine by 13 features.

The *Laryngeal* dataset comes from the Bulgarian Academy of Sciences and is available at [13]. The set was originally used for a computer decision support system, in order to aid diagnosis of laryngeal pathology and especially in detecting its early stages. Normal and pathological voices are described by 16 parameters in the time, spectral and cepstral domains.

We use a classical multidimensional scaling (MDS) for visualizing the structures in the feature space of both *Wine* and *Laryngeal* data sets (see Fig. 3 and also `mds_cs` function in [10]) .



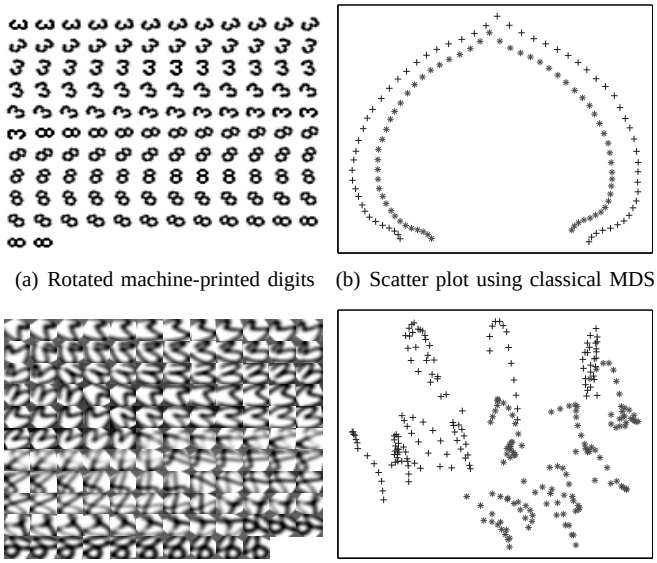
(a) *Wine* data set (b) *Laryngeal* data set

Fig. 3. Scatter plots using a classical multidimensional scaling for the *Wine* and *Laryngeal* data sets.

In order to simulate a problem with intrinsic correlations, we have created the rotated machine-printed digits shown in Fig. 4(a). In digit (machine-printed or handwritten) recognition problems, the two-class subproblem for digits “3” and “8”

is usually considered for testing the recognition ability of a particular algorithm. This two-class problem is more difficult than the other two-class subproblems because the strokes for the digits "3" and "8" are very similar. Considering that, we have taken one 16×16 example of each digit, namely "3" and "8", rotating them from -90° to 90° with steps of 3°. Even though this is a very simple problem which indeed can be corrected by using invariants [14], it is useful to illustrate the piecewise description performed by the feature lines.

The *rotated handwritten digits* are a subset of the *Digits* data, which come from the Austrian Research Institute for Artificial Intelligence [15]. Digits were downsampled to 16×16 pixels with Mitchell filter with parameter blur set to 2.5. Similarly to the machine-printed digits, we selected a subset of handwritten digits including various examples of "3" and "8", which were rotated from -60° to 60° (10° per step) as shown in Fig. 4(c).



(a) Rotated machine-printed digits (b) Scatter plot using classical MDS

(c) Rotated handwritten digits (d) Scatter plot using classical MDS

Fig. 4. Digits '3' and '8' rotated (a)-(b) between -90° and 90° with steps of 3° and (c)-(b) between -60° and 60° with steps of 10°.

B. Length-based selection of feature lines

In [7], we presented a length-based selection procedure for feature lines. In brief, it consists in ranking all the feature lines according to their length (i.e. d_{ij} in Fig. 2) and, afterwards, using such a criterion to decide if a feature line is included in R_L or not. Consider first the inclusion of the shortest feature lines, which we call the ascending selection method. In this case, the initial representation set R_L for the ascending method is the shortest feature line. Then, the second shortest feature line is added to R_L, followed by the third shortest one and so on. The reverse case corresponds to the selection in descending order. At the end, when all the n_L feature lines are included, the ascending and descending length-ranked sets are flipped versions of each other. Our conclusion in [7] was that just a few long feature lines are needed to describe correlated data sets, in comparison with the number of short feature lines required to reach a similar performance. Now, we explore a slightly different alternative. Our hypothesis is that the middle-length feature lines might be better to describe slightly non-linear subspaces, i.e. curved manifolds. So, we start the selection in the middle of the ranking. The first included feature line is that exactly placed in in the middle of the sorted list. Thereafter, taking the middle of the sorted list as reference, feature lines placed at its left and right sides are alternately included.

C. Experimental Results

Classification errors obtained for both longest and middle-length lines are shown in Figs. 5–8, as functions of the number of feature lines included in R_L. The maximum number of prototypes considered is $r_c = 15$. As a result, the total number of feature lines is 315 for the *Wine* data set (three-class problem) and 210 for the other two-class data sets. The best results obtained by the 1-NN rule and the BayesNL classifiers in the dissimilarity space are also shown in the figures. For each repetition, a new representation set R is randomly chosen. Consequently, these best results, that we used as a reference, do not necessarily correspond to the case $R = T$.

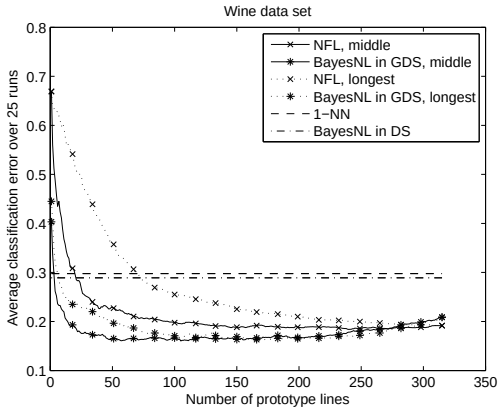


Fig. 5. *Wine* data set. Average classification errors in the generalized dissimilarity space (GDS) of the BayesNL and 1-NN classifiers. Longest and middle-length feature lines are incrementally included. Errors of the 1-NN rule and the BayesNL in the dissimilarity space (DS) are also plotted as a reference.

The first striking observation is that the BayesNL classifier in GDS outperforms both the 1-NN rule and the BayesNL in DS for the *Wine* and *Laryngeal* data sets. In contrast, for the digit recognition problems, the NFL rule outperforms the dissimilarity-based classifiers as well as the 1-NN rule. However the fact we are interested in here, the benefit of using the middle-length feature lines, is consistently observed in all the figures. Notice that the solid curves in the figures are mostly below the dotted ones. An interesting observation for the *Wine* data set is the remarkable improvement achieved when using feature lines. Since the features were not scaled before to the unit variance, we can attribute such an improvement to the capacity of feature lines for dealing better with non-scaled data. In Fig. 6, it is noteworthy that the middle-length feature lines are beneficial for the NFL rule while the longest lines provide a better description for the BayesNL classifier. The

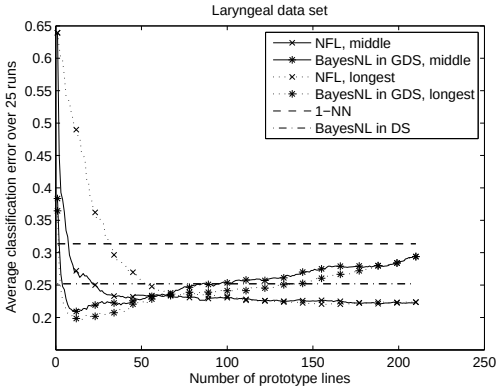


Fig. 6. *Laryngeal* data set. Average classification errors in the generalized dissimilarity space (GDS) of the BayesNL and 1-NN classifiers. Longest and middle-length feature lines are incrementally included. Errors of the 1-NN rule and the BayesNL in the dissimilarity space (DS) are also plotted as a reference.

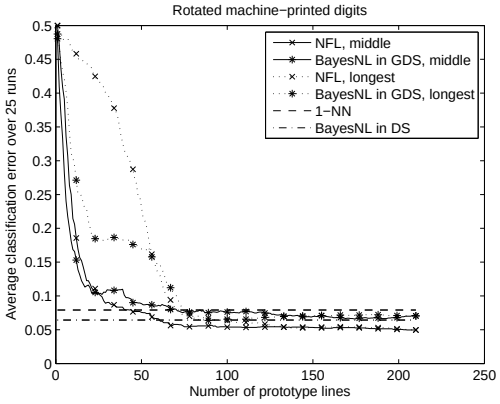


Fig. 7. *Rotated machine-printed digits*. Average classification errors in the generalized dissimilarity space (GDS) of the BayesNL and 1-NN classifiers. Longest and middle-length feature lines are incrementally included. Errors of the 1-NN rule and the BayesNL in the dissimilarity space (DS) are also plotted as a reference.

performances obtained by NFL in Figs. 7–8 and the structures presented in Figs. 4(b) and 4(d) lead us to deduce that a few middle-length feature lines may describe curved subspaces better than a small number of the longest feature lines.

IV. Conclusion

In this study we have explored the use of middle-length feature lines for generalized dissimilarity representations, compared to the results obtained by using the longest feature lines. Our experiments showed that the middle-length feature lines may provide a more accurate representation for curved subspaces than the description provided by the longest lines. Such an observation was made not just for the dissimilarity-based classifiers but also for the nearest feature line rule. The middle-length feature lines may provide a better piecewise description of the structure of the data because they are less likely to cross the territory of the other class than the longest feature lines. Besides, we can deduce that the middle-length feature lines suffer less of extrapolation and interpolation innacuracies, likely providing right directions. An apparently

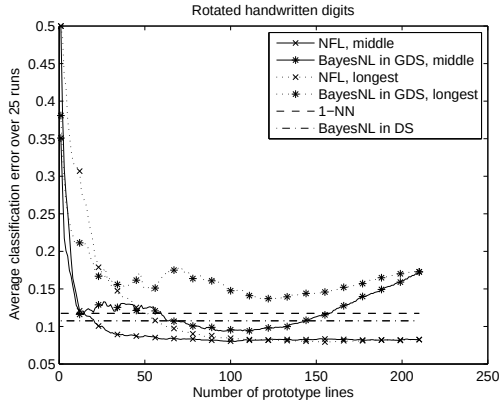


Fig. 8. *Rotated handwritten digits*. Average classification errors in the generalized dissimilarity space (GDS) of the BayesNL and 1-NN classifiers. Longest and middle-length feature lines are incrementally included. Errors of the 1-NN rule and the BayesNL in the dissimilarity space (DS) are also plotted as a reference.

promising procedure is the recent proposal by Du and Chen [9] to segment feature lines and remove those trespassing the territory of other classes. We will use this selection procedure in a our future work on classification in rectified generalized dissimilarity representations.

Acknowledgements

This work is supported by a TU Delft Research Grant and the Scholarship Program for Outstanding Postgraduate Students granted by the National University of Colombia.

References

[1] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inform. Theory*, vol. IT-13, no. 1, pp. 21–27, 1967.

[2] E. Pękalska and R. P. W. Duin, "Dissimilarity representations allow for building good classifiers," *Pattern Recognition Lett.*, vol. 23, pp. 943–956, 2002.

[3] ——, *The Dissimilarity Representation for Pattern Recognition: Foundations and Applications*. Singapore: World Scientific, 2005.

[4] S. Z. Li and J. Lu, "Face recognition using the nearest feature line method," *IEEE Trans. Neural Networks*, vol. 10, no. 2, pp. 439–443, 1999.

[5] J.-T. Chien and C.-C. Wu, "Discriminant waveletfaces and nearest feature classifiers for face recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 12, pp. 1644–1649, 2002.

[6] M. Orozco-Alzate and C. G. Castellanos-Domínguez, "Comparison of the nearest feature classifiers for face recognition," *Machine Vision and Applications (Springer)*, vol. 17, no. 5, pp. 279–285, October 2006.

[7] M. Orozco-Alzate, R. P. W. Duin, and C. G. Castellanos-Domínguez, "Generalizing dissimilarity representations using feature lines," in *submitted to the 12th Iberoamerican Congress on Pattern Recognition*, November 2007.

[8] M. Lozano, J. M. Sotoca, J. S. Sánchez, F. Pla, E. Pękalska, and R. P. W. Duin, "Experimental study on prototype optimisation algorithms for prototype-based classification in vector spaces," *Pattern Recognition*, vol. 39, no. 10, pp. 1827–1838, 2006.

[9] H. Du and Y. Q. Chen, "Rectified nearest feature line segment for pattern classification," *Pattern Recognition*, vol. 40, no. 5, pp. 1486–1497, 2007.

[10] R. P. W. Duin, P. Juszczak, D. de Ridder, P. Paclík, E. Pękalska, and D. M. J. Tax, "PRTools4: a Matlab Toolbox for Pattern Recognition," Information and Communication Theory Group: Delft University of Technology, The Netherlands, Tech. Rep., 2004, http://www.prtools.org/.

[11] E. Pękalska, R. P. W. Duin, and P. Paclík, "Prototype selection for dissimilarity-based classifiers," *Pattern Recognition*, vol. 39, no. 2, pp. 189–208, 2006.

[12] A. Asuncion and D. J. Newman, “UCI repository of machine learning databases,” 1998. [Online]. Available: http://www.ics.uci.edu/~mlearn/MLRepository.html

[13] L. I. Kuncheva, “Real medical data sets,” School of Informatics: University of Wales, Bangor, UK, Tech. Rep., 2005, http://www.informatics.bangor.ac.uk/~kuncheva/activities/real_data_full%_set.htm.

[14] J. Wood, “Invariant pattern recognition: a review,” *Pattern Recognition*, vol. 29, no. 1, pp. 1–17, 1996.

[15] A. K. Seewald, “Digits – A dataset for Handwritten Digit Recognition,” Austrian Research Institute for Artificial Intelligence, Tech. Rep. TR-2005-27, 2005. [Online]. Available: http://alex.seewald.at/digits/