Features: The More The Better

DOMINGO MERY and ALVARO SOTO
Pontificia Universidad Catolica de Chile
Department of Computer Science
Av. Vicuna Mackenna 4860(143), Macul, Santiago
CHILE
dmery,asoto@ing.puc.cl

Abstract: In pattern recognition problems, it is usually recommended to extract a low number of features in order to avoid the computational cost. However, using today's computer capabilities we are able to extract and process more features than before. In this way, in an off-line training process, it is possible to extract a very large number of features with the goal of finding relevant features for the classification task. Afterwards, in an on-line testing process, we can extract only the relevant features to classify the samples. In this paper, we use this idea to present a highly general pattern recognition methodology applied to image analysis. We combine feature extraction and feature selection techniques with highly simple classifiers to achieve high classification performances. The key idea of the proposed method is to select during training time, from a large universe of features (in some cases more than 1500 features), only those features that are relevant for the separation of the classes. We tested our methodology on six different recognition problems (with 2, 3, 6, 10 and 40 classes) yielding classification rates exceding 85% in accuracy in every case using no more than 8 features. The selected features are so robust that well known and simple classifiers are able to separate the classes.

Key–Words: Feature extraction, feature selection, classification, pattern recognition, image analysis

1 Introduction

It is well known that the automatic pattern recognition process using digital images consists of five steps [1], as shown in Fig. 1:

1. *Image Acquisition*: The digital image of the object under test is taken and stored in the computer.

2. *Image Preprocessing*: The quality of the image is improved in order to enhance its details.

3. *Image Segmentation*: The regions of interest of the image are found and isolated from the rest of the scene.

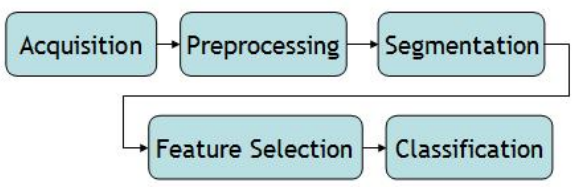

Figure 1: Pattern recognition schema used in image analysis.

4. *Feature Extraction*: The regions are measured and some significant characteristics are quantified.

5. *Classification*. The extracted features of each region are analyzed and assigned to one of the defined classes.

Traditionally, it is recommended to extract a low number of features in order to avoid the computational cost [15, 3]. However, using today's computer capabilities we are able to extract a very large number of features in a off-line process in order to investigate which features are really relevant. Thus, in a on-line process we can extract only the relevant features to classify the samples.

In this paper we present a nobel pattern recognition methodology, in which we use a very large number of features combined with a feature selection approach to perform very good classifications. The key idea of the proposed method is to select, from a large universe of features (in some cases more than 1500 features), only those features that are relevant for the separation of the classes. We tested our methodology on six different recognition problems using only visual data, but we believe that the same methodology can be used with other kind of data. The selected features are so robust that well known and simple clas-

sifiers are able to separate the classes. In our experiments, we evaluate the performance in cases with 2, 3, 6, 10 and 40 classes. The classification performance was over 85% in every case using no more than 8 features, and in some cases over 95% using only 4 features.

The rest of the paper is organized as follows: In Section 2 we explain the feature extraction used in our approach. In Section 3 we give some details of the feature selection and classification. In Section 4 we show the experimental results. Finally, in Section 5 we give some concluding remarks.

2 Feature extraction

In this Section we concentrate on the extraction of features, whereas in the next Section we will discuss the feature selection and classification problems. In our description, features will be divided into two groups: *geometrical* and *intensity* features (see Fig. 2).

Geometrical features provide information on the size and shape of a segmented region. Size features, such as area, perimeter, height and width, are given in pixels. Shape features are usually attributed coefficients without units. In our approach, we extract the following four groups of features:

1. Standard geometrical features using the command `regionprops` [8], where area, orientation, Euler number, solidity among others are computed.

2. Invariant features like Hu moments [5, 14] that are invariant under magnification, translation and rotation.

3. Fourier Descriptors because they may also be good choice for establishing the shape [13].

4. Elliptical features obtained from a fitted ellipse to the boundary of the region [2].

Totaly, we extract 54 geometrical features.

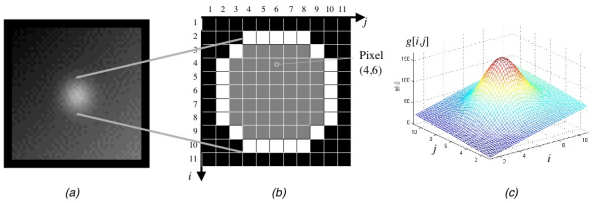

Figure 2: Example of a region: a) image, b) segmented region used for geometrical features, c) intensity values used for intensity features.

Intensity features provide information about the color intensity of a segmented region. These features can be extracted for each intensity channel, e.g., gray value, red, green, blue, hue, saturation, value, etc. We extract the following six groups of features:

1. Standard intensity features are related to the mean, standard deviation of the intensity in the region, mean first derivative in de boundary, and second derivative in the region [11].

2. Contrast features provide information of the intensity difference between a region and its neighborhood [10].

3. Haralick and Gupta texture features take into account the distribution of the intensity values in the region [4, 14], where mean and range of the following variables are measured: Angular Second Moment, Contrast, Correlation, Sum of squares, Inverse Difference Moment, Sum Average, Sum Entropy, Sum Variance, Entropy, Difference Variance, Difference Entropy, Information Measures of Correlation, and Maximal Correlation Coefficient.

4. Fourier and discrete cosine transform coefficients [1].

5. Hu moments with intensity information [5].

6. Gabor features based on 2D Gabor functions, i.e., Gaussian-shaped bandpass filters, with dyadic treatment of the radial spatial frequency range and multiple orientations, which represent an appropriate choice for tasks requiring simultaneous measurement in both space and frequency domains [7] (we use 8 scale and 8 orientations).

Totally, we extract 227 features per channel, i.e., for a color image we can extract these features for red, green and blue channels ($3 \times 227 = 681$ features) and 681 more features if we want to analyze the HSV color space.

3 Feature selection and classification

In feature selection we have to decide just which features of the regions are relevant for the classification task at hand. The n extracted features are arranged in an n-vector: $\mathbf{w} = [w_1 ... w_n]^\mathsf{T}$ that can be viewed as a point in a n-dimensional feature space. The features are normalized as

$$\tilde{w}_{ij} = \frac{w_{ij} - \bar{w}_j}{\sigma_j} \qquad (1)$$

for $i = 1, ..., N_0$ and $j = 1, ..., n$, where w_{ij} denotes the j-th feature of the i-th feature vector, N_0 is the number of samples, and $\bar{w_j}$ and σ_j are the mean and standard deviation of the j-th feature. The normalized features have zero mean and a standard deviation equal to one.

The key idea of the feature selection is to select a subset of m features ($m < n$) that leads to the smallest classification error. The selected m features are arranged in a new m-vector $\mathbf{z} = [z_1 ... z_n]^\mathsf{T}$. The selection of the features can be done using *Sequential Forward Selection* (SFS) [6]. This method selects the best single feature and then adds one feature at a time that, in combination with the selected features, maximizes classification performance. The iteration is stopped once no considerable improvement in the performance is achieved on adding a new feature. By evaluating selection performance we ensure: *i)* a small intraclass variation and *ii)* a large interclass variation in the space of the selected features. For the first condition the intraclass-covariance is used:

$$\mathbf{C}_b = \sum_{k=1}^{N} p_k (\bar{\mathbf{z}}_k - \bar{\mathbf{z}})(\bar{\mathbf{z}}_k - \bar{\mathbf{z}})^\mathsf{T}, \qquad (2)$$

where N means the number of classes, p_k denotes the a-priori probability of the k-th class, $\bar{\mathbf{z}}_k$ and $\bar{\mathbf{z}}$ are the mean value of the k-th class and the mean value of the selected features. For the second condition the interclass-covariance is used:

$$\mathbf{C}_w = \sum_{k=1}^{N} p_k \mathbf{C}_k, \qquad (3)$$

where the covariance matrix of the k-th class is given by:

$$\mathbf{C}_k = \frac{1}{L_k - 1} \sum_{j=1}^{L_k} (\mathbf{z}_{kj} - \bar{\mathbf{z}}_k)(\mathbf{z}_{kj} - \bar{\mathbf{z}}_k)^\mathsf{T}, \qquad (4)$$

with $\mathbf{z}_{kj}$ the j-th selected feature vector of the k-th class, L_k is the number of samples in the k-th class. Selection performance can be evaluated using the spur criterion for the selected features $\mathbf{z}$:

$$J = \text{spur}\left(\mathbf{C}_w^{-1} \mathbf{C}_b\right). \qquad (5)$$

The larger the objective function J, the higher the selection performance. For more details see [15].

Once the proper features are selected, a classifier can be designed. The classifier assigns a feature vector $\mathbf{z}$ to one of the determined classes. In statistical pattern recognition, classification is performed using the concept of similarity, where *similar* patterns are assigned to the same class [6]. Although this approach is very simple, a good metric defining the similarity must be established. Using a representative sample we can make a supervised classification finding a discriminant function $d(\mathbf{z})$ that provides us with information about how similar a feature vector $\mathbf{z}$ is to the feature vector of a class.

4 Experimental Results

In order to test our methodology, we used the following six different sets of data:

Set 1 *Face recognition*: In this set there are 10 different frontal pictures from 40 persons. The images are very small (56 $\times$ 45 pixels) in gray values. The idea is to recognize the person from her/his picture.

Set 2 *Digit recognition*: In this set there are approximately 30 images for each digit (0...9) in different sizes and different orientations. The binary images are small (from 14 $\times$ 28 to 89$\times$ 149 pixels). The idea is to recognize the digits.

Set 3 *Potato chip quality recognition*: In this set there are 10 different color pictures from 6 different qualities of potato chips. The images are large (1536 $\times$ 2048 pixels). The idea is to recognize the quality.

Set 4 *Tortilla quality recognition*: In this set there are 100 different color pictures from 3 different qualities of tortillas. The images are very large (2304 $\times$ 3072 pixels). The idea is to recognize the quality.

Set 5 *Face detection*: In this set there are 264 different small pictures (64 from frontal faces and 200 randomly chosen from digital pictures with no faces). The color images are small (from 50 $\times$ 50 to 300$\times$ 300 pixels). The idea is to detect if there is a frontal face in the pictures.

Set 6 *Gender detection*: In this set there are 610 different small frontal pictures of persons (218 females and 392 males). The color images are small (211 $\times$ 117 pixels). The idea is to detect the gender of the person.

In each set we extracted a very large number of features. Depending on the data, geometrical and several intensity features were extracted (e.g., in set 2 only geometrical features were extracted because the input data were binary images and the relevant information for the class separation was in the shape only).

After the feature extraction, we selected randomly 75% of the samples of each class to perform the feature selection. In this step we eliminated the constant features (e.g., Euler number in regions with no holes, this is the case of Set 3 and 4), and the high correlated features (e.g., intensity features extracted from hue channel and gray value). Afterwards, we normalized features using the linear transform (1) in order to obtain features with zero mean and a standard deviation equal to one. Finally, we selected the best 20 features using SFS (see an example in Fig. 3 where we can see how objective function J from (5) is maximized).

We selected the first p = 1, 2, 4, 8, and 20 features obtained by SFS and we trained two well known classifiers: Probabilistic Neural Network (PNN) [9] and Linear Discriminant Analysis (LDA) [15]. More sophisticated classification techniques can always be used; however, in this article we show that a simple classification technique is good enough to distinguish among the classes.

The performance was evaluated using cross-validation, a technique widely used in machine learning problems [12]. In cross-validation, the training and testing process is repeated several times to test the stability of the classifier. Then, when training is performed, the samples that were initially removed can be used to test the performance of the classifier on these test data. Thus, one can evaluate how well the method will classify samples that have not already examined. In our experiments, 80% of the whole data was used to train and the rest (20%) for test. We repeated this experiment 5 times rotating train and test data. The mean of the 5 percentages of the true classifications were tabulated in each case as shown in Table 1. In this table %PNN-p (or %LDA-p) means that the classifier PNN (or LDA) was used with the first p selected features of SFS. An interesting result can be observed in Set 5 where we are able to detect a face in 90% of the cases using only two features. This result is illustrated in Fig. 4 where the separability of the two classes is evident.

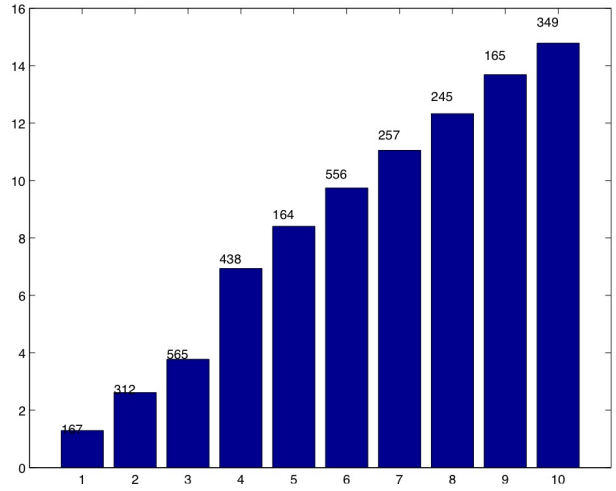

Figure 3: Sequential Forward Selection for the first 10 features of Set 5.

We can see that in every dataset our methodology is able to perform a good classification. Using 8 features we can obtain a performance exceding 85% in every set. Additionally, in two cases (Set 4 and 5) the performance is 95% or more using only 4 features.

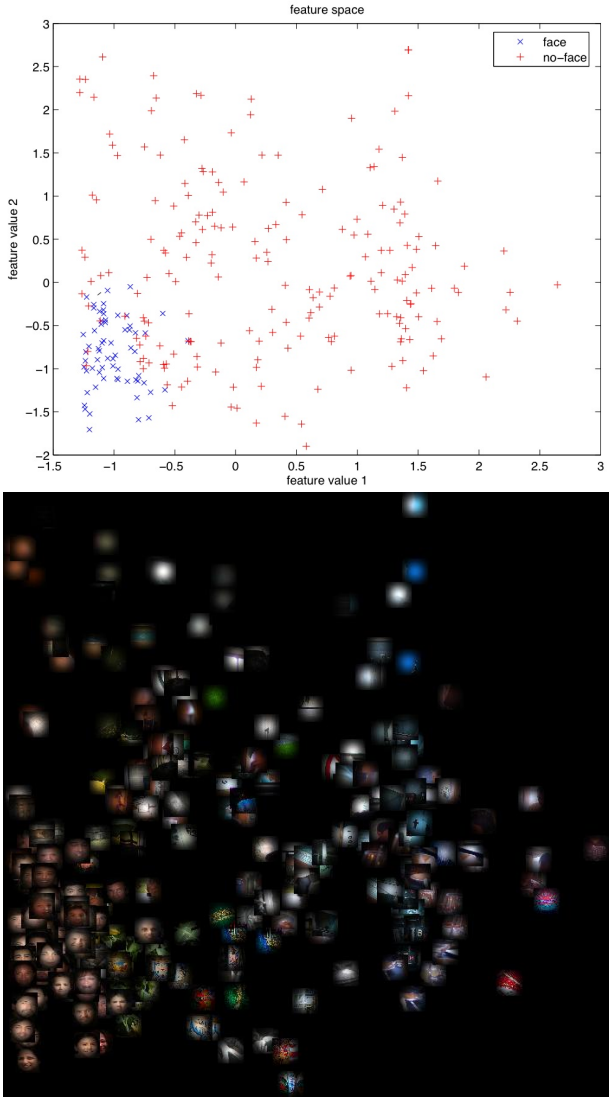

Figure 4: Representation of the best two features in detection of faces (Set 5). Top: feature space. Bottom: original face and no-face images superimposed in the feature space. We observe that the faces are located bottom left.

Table 1: Performance of the method.

Set →	1	2	3	4	5	6
Gray values	yes	no	yes	yes	yes	yes
RGB Color	no	no	yes	yes	yes	yes
HSV Color	no	no	yes	yes	yes	yes
Shape	no	yes	yes	yes	no	no
Classes	40	10	6	3	2	2
Samples	400	307	60	300	264	610
Features	227	54	1643	1643	1589	1589
%PNN-1	5	31	33	72	76	71
%PNN-2	14	36	63	100	90	78
%PNN-4	31	49	70	100	96	80
%PNN-8	77	76	83	100	98	95
%PNN-10	84	76	72	100	98	97
%PNN-20	94	87	78	100	98	97
%LDA-1	16	62	62	97	76	68
%LDA-2	33	62	65	100	82	73
%LDA-4	56	85	83	100	94	84
%LDA-8	86	84	90	100	96	89
%LDA-10	87	86	92	100	96	89
%LDA-20	97	91	90	100	98	95

5 Conclusion

The results explained in the previous sections show that using a very large number of features combined with a feature selection approach allow us to achieve high classification rates on a wide variety of visual classification tasks. This demonstrates the generality of the proposed methodology. The key idea of the proposed method is to select, from a large universe of features, only those features that are relevant for the separation of the classes. We tested our method in six different recognition problems (with 2, 3, 6, 10 and 40 classes) yielding a performance over 85% in accuracy for every case using no more than 8 features. We believe that the proposed methodology opens up new possibilities in the field of image analysis and pattern recognition.

Acknowledgements: The research was partially supported CONICYT-Chile project ACT-32 and partially supported by a grant from the School of Engineering at Pontificia Universidad Catolica de Chile.

References:

[1] K.R. Castleman. *Digital image processing*. Prentice-Hall, Englewood Cliffs, New Jersey, 1996.

[2] A. Fitzgibbon, M. Pilu, and R.B. Fisher. Direct least square fitting ellipses. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(5):476–480, 1999.

[3] K. Fukunaga. *Introduction to statistical pattern recognition*. Academic Press, Inc., San Diego, 2 edition, 1990.

[4] R.M. Haralick. Statistical and structural approaches to texture. *Proc. IEEE*, 67(5):786–804, 1979.

[5] M.-K. Hu. Visual pattern recognition by moment invariants. *IRE Trans. Info. Theory*, IT(8):179–187, 1962.

[6] A.K. Jain, R.P.W. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(1):4–37, 2000.

[7] A. Kumar and G.K.H. Pang. Defect detection in textured materials using gabor filters. *IEEE Transactions on Industry Applications*, 38(2):425–440, 2002.

[8] MathWorks. *Image Processing Toolbox for Use with MATLAB: User's Guide*. The MathWorks Inc., January 2003.

[9] MathWorks. *Neural Network Toolbox for Use with MATLAB: User's Guide*. The MathWorks Inc., January 2006.

[10] D. Mery. Crossing line profile: a new approach to detecting defects in aluminium castings. *Lecture Notes in Computer Science*, 2749:725–732, 2003.

[11] D. Mery and D. Filbert. Classification of potential defects in automated inspection of aluminium castings using statistical pattern recognition. In 8^{th} *European Conference on Non-Destructive Testing (ECNDT 2002)*, Barcelona, 17-21 June 2002.

[12] T.M. Mitchell. *Machine Learning*. McGraw-Hill, Boston, 1997.

[13] E. Persoon and K.S. Fu. Shape discrimination using Fourier descriptors. *IEEE Trans. Systems, Man, and Cybernetics*, SMC-7(3):170–179, 1977.

[14] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. PWS Publishing, Pacific Grove, CA, 2 edition, 1998.

[15] A. Webb. *Statistical Pattern Recognition*. Wiley, England, 2005.