International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 18, No. 5 (2004) 819–845

World Scientific
www.worldscientific.com

AUTOMATIC CLASSIFICATION OF DIGITAL PHOTOGRAPHS BASED ON DECISION FORESTS

RAIMONDO SCHETTINI
DISCo, University of Milano Bicocca,
Via Bicocca degli Arcimboldi 8, Milano, 20126, Italy
schettini@disco.unimib.it

CARLA BRAMBILLA
IMATI, CNR, Via Bassini 15, Milano, 20133, Italy
carla@mi.imati.cnr.it

CLAUDIO CUSANO* and GIANLUIGI CIOCCA†
ITC, CNR, Via Bassini 15, Milano, 20133, Italy
DISCo, University of Milano Bicocca,
Via Bicocca degli Arcimboldi 8, Milano, 20126, Italy
**cusano@disco.unimib.it*
†*ciocca@disco.unimib.it*

Annotating photographs with broad semantic labels can be useful in both image processing and content-based image retrieval. We show here how low-level features can be related to semantic photo categories, such as indoor, outdoor and close-up, using decision forests consisting of trees constructed according to CART methodology. We also show how the results can be improved by introducing a rejection option in the classification process. Experimental results on a test set of 4,500 photographs are reported and discussed.

Keywords: CART; decision forest; digital images; image classification; low-level features.

1. Introduction

The automatic classification of digital photographs in semantic categories is an unresolved challenge in the multimedia and imaging communities. The demands of a number of practical applications call for further research on this problem, one of which concerns the optimization of image processing. For example, if printers, scanners, photocopiers and fax machines could take into account the content of the photograph, they could automatically adopt the most appropriate strategies of color adjustment. Another important application concerns image retrieval. The successful semantic classification of photographs could greatly enhance the performance of retrieval systems by filtering out photographs from irrelevant classes during the

matching phase and guiding the identification of specific objects present in the image.[26]

Low-level features, that is, features that can be automatically computed without any *a priori* knowledge of the content of the image, have recently been used with promising results to index images in the automatic classification of digital images. But very little work to date has concerned digital photographs.

Athitsos and Swain[2] and Gevers *et al.*[12] studied the problem of distinguishing between photographs and graphics, Schettini *et al.*[22,23] that of sorting photographs, graphics, texts and compound documents.

Lienhart and Hartmann[18] presented algorithms for distinguishing photo-like images from graphical images, real photographs from photo-like images, and artificial images and presentation slides/scientific posters from comics.

Yiu classified pictures as indoor or outdoor scenes using color histograms and texture orientation. She reported an accuracy of 92% on a database of about 500 images, obtained by stacking k-nearest neighbors and support vector machine classifiers.[34] Szummer and Picard[30] classified photographs as indoor or outdoor by using a k-nearest neighbors classifier. To classify an image, they computed the features on subblocks, classified the subblocks independently, and then combined the results. Three types of features were employed, one each for color, texture and frequency distribution. The color feature used was a color histogram computed on the Otha color space, while texture features were multiresolution, autoregressive model parameters, and frequency features were DCT coefficients. The highest accuracy reported was 90.3%, evaluated on a database of over 1,300 images.

Vailaya *et al.*[32] worked on the problem of distinguishing between city images and landscapes. They have used a weighted k-nearest neighbors classifier, which provided an accuracy of 93.3% on a database of about 2,700 images. The features considered were: color histograms and coherence vectors in the HSV space, DCT coefficients, edge direction histograms and coherence vectors. Finally, Vailaya *et al.*[31] considered the hierarchical classification of vacation images: at the highest level the images were sorted into indoor/outdoor classes, outdoor images were then assigned to city/landscape classes, and, finally, landscape images were classified in sunset, forest and mountain categories. Vector quantization was used to estimate the class-conditional densities of the features and then derive a maximum *a posteriori* probability criterion. At the top level of the hierarchy, that is for indoor/outdoor classification, this system achieved an accuracy of 90.5% on a database of about 7,000 images. At the bottom line, for classification as sunset, forest or mountain, the accuracy was 96.6%. Different features were used, depending on the specific problem addressed: color moments in the LUV space for indoor/outdoor classification, edge direction histograms and coherence vectors for city/landscape classification, spatial moments, color histograms and coherence vectors in the HSV and LUV spaces for sunset/forest/mountain classification.

We present here our experimentation on indoor/outdoor/close-up classification based on the use of ensembles of decision trees, often called decision forests.[9,15] The

trees of the forests have been constructed according to CART methodology.[3] The classes taken into account correspond to typologies of images that require different enhancement approaches in the image processing chain. In the framework of image retrieval these classes could be further split, as in Ref. 31.

We have chosen CART trees as classifiers because they are very powerful in handling nonhomogeneous relationships among the predictors, and this, we feel, is of primary importance in problems characterized by a high level of complexity. In the unsolved problem of distinguishing among indoor, outdoor, and close-up images, a precise mapping from a set of low-level features to image semantics is not possible, and the assumption that images with similar semantics share the same pictorial characteristics may not always hold.

The features we use are related to color (moments of inertia of the color channels in the HSV color space, and skin color distribution), texture and edge (statistics on wavelets decomposition and on edge and texture distributions), and composition of the image (in terms of fragmentation and symmetry). To fully exploit the fact that trees allow a powerful use of high dimensionality and conditional information, we take all the features together and let the training process perform complexity reduction, and detect any redundancy. Our approach differs in this from that of the authors cited above, who tested the performance of one feature, or a combination of a few features, at a time.

We have also included a rejection option[13,17] in the classification process, precisely an ambiguity rejection option.[33] This has several advantages: (i) images that are to be classified are not compulsorily assigned to one of the designated classes; (ii) ambiguous images, that is, images that may be labeled differently by different observers, such as an indoor picture of a window, are likely to be rejected; and (iii) classification accuracy for nonambiguous images is improved.

The paper is organized in five sections. Section 2 outlines CART methodology, describes how we built the decision forests, and the rejection rules we experimented in order to implement the rejection option. Section 3 illustrates the features used for image indexing, while Sec. 4 reports the results obtained on a test set of 4,500 images, extracted from a database of 9,000 images collected from various sources. In Sec. 5, we present our conclusions, with a road map for the future.

2. Classification Methodology

Our classification process is based on the use of decision trees built according to CART methodology.[3] These are binary trees produced by recursively partitioning the predictor space, each split being formed by conditions related to the predictor values. Each subset corresponds to a node of the tree: the whole predictor space corresponds to the root node, the subsets of the final partition correspond to the terminal nodes. Once a tree has been constructed, a class is assigned to each of the terminal nodes, and it is this that makes the tree a classifier: when a new case is processed by the tree, the class associated with the terminal node in which the case ends up on the basis of its predictor values is its predicted class.

In problems where it is feasible to assume that the cost of misclassifying a class j case as a class i case is the same for all $i \neq j$, $i, j = 1, \ldots, J$, the class assigned to each terminal node t is the class i for which $p(i|t) = \max_j p(j|t)$, where $p(j|t)$ is the resubstitution estimate of the conditional probability of class j in node t, that is, the probability that a case found in node t is a class j case. With this rule the resubstitution estimate of the accuracy inside the node, given by $p(i|t)$, is maximized or, equivalently, the resubstitution estimate of the misclassification probability inside the node, given by $1-p(i|t)$, is minimized. If the prior probabilities of the classes are estimated from the data, $p(i|t)$ is simply the proportion of class i cases inside node t and the resubstitution estimate of the accuracy inside the node is reduced to the relative proportion of cases in the node that belong to class i.

When it is not realistic to assume equal misclassification costs, the class assigned to each terminal node of the tree is the class for which the estimated misclassification cost inside the node is minimized. In our study we have assumed equal misclassification costs.

The critical problems of the splitting process are essentially two: how to identify candidate splits, and how to define the goodness of the splits. Candidate splits are generated by a set of admissible questions regarding the values of the predictors, which differ according to the nature of the predictors themselves. At each step of the process, all the predictors are searched one by one, and the best split, in the sense defined below, is found for each predictor. The best splits are then compared, and the best of these selected.

The idea central to the goodness of splits is that of selecting the splits so that the data in the descendant nodes are purer than the data in the original ones. To do so, a function of impurity of the nodes, $i(t)$, is introduced, and the decrease in its value produced by a split is taken as a measure of the goodness of the split itself.

The function of node impurity we have used is the Gini diversity index

$$i(t) = \sum_{i \neq j} p(i|t)p(j|t) = 1 - \sum_j p^2(j|t)\,, \tag{1}$$

which has a clear interpretation in terms of variances of Bernoulli variables. If, for each class j, we consider the random variable Y_j, which is 1 (success) if a case of t belongs in class j and 0 (failure) otherwise, it can be modeled as a Bernoulli variable whose probability of success is estimated by $p(j|t)$, and the quantity

$$1 - \sum_j p^2(j|t) \tag{2}$$

is the sum of the estimated variances of such variables.

In CART methodology the size of a tree is treated as a tuning parameter, and the optimal size is adaptively chosen from the data. A very large tree is grown and then pruned, using a cost-complexity criterion which governs the tradeoff between size and accuracy, or cost. This eliminates both the risk of large trees which overfit the training data, as well as that of small trees that do not capture important information. The pruning process generates a sequence $\{T_l\}_{l \in \{1,\ldots,L\}}$ of subtrees

decreasing in size; these are evaluated in terms of their accuracy, or misclassification cost, and the best subtree is then selected. When large sets of data are available, as is the case here, the accuracy, or misclassification cost, of the subtrees are usually estimated on the basis of a test set. Otherwise, cross-validation must be applied.

Although the pruning process prevents the danger of trees too tailored to the training data, there is still overfitting due to instability, a phenomenon inherent in the hierarchical nature of the construction process of trees. Even a small change in data may result in a very different series of splits, and this clearly affects both the structure of the trees, and the consequent classification results.

The use of decision forests has proved quite successful in overcoming the above problem, and in improving generalization accuracy.[8,14] The different trees of a forest are generated by manipulating the training set, and running the training process on the different sets thus obtained. Classification results are then combined in various ways. We have chosen this approach, and used bagging, one of the most effective computationally intensive methods for improving unstable classifiers.[4,5]

With bagging, trees are formed by making bootstrap replicates of the training set, and using these as new training sets. In any particular bootstrap replicate, each element of the training set may appear a number of times, or not at all, since the replicates are obtained by resampling with replacement. To combine the classification results produced by the single trees we have appplied the majority vote rule. Boosting is another way of deriving decision forests.[14,17]

To avoid doubtful decisions, we decided to integrate an ambiguity rejection option[33] in the classification process. To do so, we experimented with two different rejection rules, which we have called the global and the local rule, respectively. The global rule states that a classification result obtained by means of the majority vote rule is rejected if the percentage of trees of the forest that contributes to it is less than a given threshold. The rule is constant over the feature space.

The local rule, instead, incorporates local information, namely, the resubstitution estimates of the accuracy within the terminal nodes of the trees. This rule operates in two steps. At first, the classification produced by any single tree of the forest is rejected if the image to be classified ends up in a terminal node with an estimated accuracy lower than a given threshold. In this case the image is temporarily assigned to a rejection class. In the second step, the classification based on the majority vote is rejected if the most voted class for the image being processed is the rejection class.

3. Features

The great variety of feature sets used in similar studies[2,12,18,30–32,34] proves indirectly that there is no single "best" representation of the content of an image, but only multiple representations which characterize the content from different perspectives.[24] No single low-level feature allows the univocal identification of an image class: images which are completely different in terms of a feature may belong

to the same class, and images which are very similar may actually belong to different classes. We have used a rich image description, in terms of color, edge, texture, and image composition, to limit the risk that images with different semantics share the same description in terms of low-level features.

The features used are described in detail in the following.

3.1. *Color distribution*

The color distribution of an image is usually characterized by the use of color histograms. However, two other approaches more efficient than those based on color histograms have been proposed,[28,29] as they do not require color quantization, and produce more compact indices. One uses only the first three moments of the color distribution of each color channel; in the other, the image is represented only by the means and covariance matrix of its color distribution. We have followed the first approach, as it is more suitable for evaluating similarities. The first three moments, mean (E), standard deviation (σ) and skewness (s), have been computed for each color channel of the HSV color space, that separates the hue component from the saturation and value components. They are:

$$E_c = \frac{1}{N} \sum_{x,y} p_c(x,y)\,, \tag{3}$$

$$\sigma_c = \sqrt{\frac{1}{N} \sum_{x,y} (p_c(x,y) - E_c)^2}\,, \tag{4}$$

$$s_c = \sqrt[3]{\frac{1}{N} \sum_{x,y} (p_c(x,y) - E_c)^3}\,, \tag{5}$$

where N is the number of pixels of the image, $c = \mathrm{H,S,V}$ refers to the color channels and $p_c(x,y)$ is the value on color channel c of the pixel at position (x,y).

3.2. *Skin detector*

To detect the presence of human beings in the image we have used the statistical skin color detector proposed by Miyake *et al.*[20] These authors have analyzed the distributions of the r, g chromaticities of the pixels of 4,000 skin image regions, and then devised a probability ellipse rule to model the chromaticities. The r, g values are computed as follow:

$$r = \frac{R}{R+G+B}\,, \tag{6}$$

$$g = \frac{G}{R+G+B}\,, \tag{7}$$

where R, G and B are the red, green and blue values of the pixels.

The conditional probability of the chromaticities of pixels belonging to the skin class S is modeled by a bivariate normal distribution. Let $\underline{c}_{xy}$ be the vector of the chromaticities of the pixel at position (x, y), then:

$$f(\underline{c}_{xy}|S) = \frac{e^{-\frac{1}{2}(\underline{c}_{xy}-\underline{\mu}_S)^T\Sigma_S^{-1}(\underline{c}_{xy}-\underline{\mu}_S)}}{2\pi|\Sigma_S|^{\frac{1}{2}}}, \tag{8}$$

where f denotes the density function.

With this assumption the quadratic form

$$U(\underline{c}_{xy}; \underline{\mu}_S, \Sigma_S) = (\underline{c}_{xy} - \underline{\mu}_S)^T\Sigma_S^{-1}(\underline{c}_{xy} - \underline{\mu}_S) \tag{9}$$

has a χ_2^2 probability distribution; therefore, given a confidence value of $\alpha \in [0, 1]$, it is possible to select the elliptic region

$$U_\alpha(\underline{c}_{xy}; \underline{\mu}_S, \Sigma_S) < \lambda_\alpha\,, \tag{10}$$

which includes, with probability α, the skin tone colors. The confidence value α is used to select the size of the elliptic region and thus define the range of the chromaticities that should be considered to belonging to the set of skin chromaticities.

We have estimated μ_S and Σ_S using a training set of 30,000 color skin data of three different human races: African, Caucasian and Indian. The skin detector feature is defined as the percentage of pixels whose cromaticities belong to the elliptic region $U_{0.75}$.

3.3. *Edge distribution*

To describe the edge distribution of the images we have used the statistical information on image edges extracted by Canny's algorithm. The Canny operator[6] is a multistage process that works on the image converted to gray scale. We have used this algorithm since it produces as output an image with edges one-pixel wide, while allowing us, at the same time, to compute a number of statistics that can be used to index the image content.

The Canny algorithm is applied to the luminance image that is smoothed by a 3×3 Gaussian filter with $\sigma = 0.01$. The algorithm initially generates two images corresponding to the gradient magnitude and orientation at each pixel. It then uses nonmaximum suppression which, by keeping only edge strength values greater than neighboring strengths, thins edges to one-pixel wide contours. Hysteresis thresholding is used to eliminate edges caused by noise, and to group the edges into numbered contours. The hysteresis phase uses two thresholds (T_1 and T_2 with $T_1 > T_2$), preserving contours that have at least one point with an edge strength over the high threshold value (T_1) and strengths over the low threshold (T_2) for the rest of the contour.

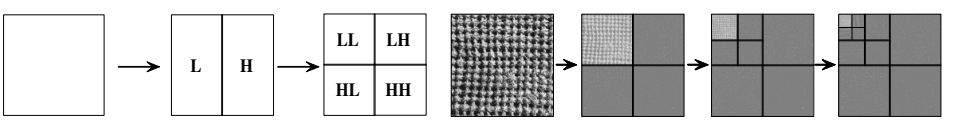
T_1 and T_2 are computed by analyzing the cumulative histogram of the strength values found in the nonmaximum suppression phase. T_1 corresponds to the strength when the histogram reaches the 0.85 value, and T_2 corresponds to 60% of the

T_1 strength. We have called the two threshold HighThreshold and LowThreshold. These, together with the following measures, are the edge features used for indexing the images:

- the percentage of possible edge pixels, that is, the percentage of nonzero pixels after the nonmaximal suppression process;
- the percentages of possible edge pixels that lie below the T_2 threshold (Low Edge), between the T_2 and T_1 thresholds (Medium Edge), and over the T_1 threshold (High Edge);
- the percentages of Medium Edge pixels not selected as true edge pixels after hysteresis (Noisy Edge) and of those promoted as true edge pixels (Promoted Edge);
- the percentage of possible edge pixels globally selected as true edge pixels (High Edge and Promoted Edge).
- the number of closed contours and the maximum gradient value;
- the histogram of edge directions quantized in 18 bins.

3.4. *Wavelets*

Multiresolution wavelet analysis provides representations of image data in which both spatial and frequency information are present.[16] In multiresolution wavelet analysis we have four bands for each level of resolution resulting from the application of two filters, one low-pass (L) and the other high-pass (H). The filters are applied in pairs in the four combinations, LL, LH, HL and HH. A decimation phase that halves the resulting image size follows. The final image, of the same size as the original, contains a smoothed version of the latter (LL band) and three bands of details [see Fig. 1(a)]. Each band corresponds to a coefficient matrix which can be used to reconstruct the original image. In our procedure the features have been extracted from the luminance image, applying a three-step Daubechies multiresolution wavelet decomposition using 16 coefficients and producing ten sub-bands[25] [Fig. 1(b)]. Two energy features, the mean E and the standard deviation σ, are then computed for each sub-band. Let D_n be the nth sub-band, $n = 1, \ldots, 10$, and



(a) (b)

Fig. 1. (a) The filtering and decimation of the image along the horizontal and vertical directions. Four bands a quarter of the size of the whole image are created. (b) The tree-step application of the multiresolution wavelet. The wavelet filters are applied to the top left band containing the resized image.

N_n its number of pixels, the two features are:

$$E_n = \frac{\sum_{i \in D_n} \sum_{j \in D_n} |D_n(i,j)|}{N_n}, \tag{11}$$

$$\sigma_n = \sqrt{\frac{\sum_{i \in D_n} \sum_{j \in D_n} (|D_n(i,j)| - E_n)^2}{N_n}}. \tag{12}$$

3.5. *Texture*

Texture features are features related to the local spatial changes in the intensity of the image pixels. We have based their computation on the Neighborhood Gray Tone Difference Matrix (NGTDM) proposed by Amadasun *et al.*[1] In this matrix each element is the absolute difference between the intensity of a pixel and the average intensity of the surrounding pixels, in a neighborhood of fixed radius (always set at 1 in our experiments). By adding all the elements of the NGTDM matrix corresponding to pixels of intensity i, we obtain the quantities $s(i)$ which are the basis for our computation of the texture features.

The features computed are those related to five texture properties: coarseness, contrast, busyness, complexity and strength. If we denote by p_i the probability of the occurrence of intensity i in the image, by G_h the highest intensity value present in the image, by N_g the number of intensity levels in the image, and by n the number of pixels left after excluding the pixels belonging to an external border of fixed width, they are computed as follows:

3.5.1. *Coarseness*

$$f_{\cos} = \left(\epsilon + \sum_{i=0}^{G_h} p_i s(i) \right)^{-1}, \tag{13}$$

where ϵ (set at 10^{-12}) is included to prevent the measure from becoming infinite. The coarseness measure provides a gauge on the granularity of the texture: high values of coarseness imply that the texture possess large patches of uniform intensities.

3.5.2. *Contrast*

$$f_{\text{con}} = \left(\frac{1}{N_g(N_g-1)} \sum_{i=0}^{G_h} \sum_{j=0}^{G_h} p_i p_j (i-j)^2 \right) \cdot \left(\frac{1}{n^2} \sum_{i=0}^{G_h} s(i) \right). \tag{14}$$

A texture presents high contrast if the differences in intensity levels (bright/dark) in neighboring regions are clearly visible.

3.5.3. *Complexity*

$$f_{\text{com}} = \sum_{i=0}^{G_h} \sum_{j=0}^{G_h} \left(\frac{|i-j|}{n^2(p_i + p_j)} \right) (p_i s(i) + p_j s(j)) \,. \tag{15}$$

A texture is complex if the information content is high, that is, the texture exhibits different regions with different average intensities.

3.5.4. *Strength*

$$f_{\text{str}} = \left(\sum_{i=0}^{G_h} \sum_{j=0}^{G_h} (p_i + p_j)(i-j)^2 \right) \cdot \left(\epsilon + \sum_{i=0}^{G_h} p_i s(i) \right)^{-1} . \tag{16}$$

A texture is strong if the texture's primitives are clearly visible.

3.5.5. *Business*

$$f_{\text{bus}} = \frac{\sum_{i=0}^{G_h} p_i s(i)}{\sum_{i=0}^{G_h} \sum_{j=0}^{G_h} (ip_i - jp_j)^2} \,. \tag{17}$$

A texture is busy if there are abrupt changes of intensity between a pixel and its neighbors, that is, the spatial frequency of intensity changes is high.

3.6. *Image composition*

In order to compute the features related to image composition, the HSV color space was partitioned into eleven color zones corresponding to basic color names, as shown in Table 1. This partition was defined and validated empirically by different groups of examiners.[10] Since after quantization the image presents noisy points due to the nonuniform color regions, a max filter with a 5×5 pixels window was applied to the segmented image to remove these small regions.

Table 1. The eleven color zones used to quantize the HSV color space.

$H_{\min}$	$H_{\max}$	$S_{\min}$	$S_{\max}$	$V_{\min}$	$V_{\max}$	Color
0	360	0	15	0	31	Black
0	360	0	15	32	69	Gray
0	360	0	15	70	100	White
−18	18	16	100	32	100	Red
19	40	16	100	32	100	Orange
41	62	16	100	32	100	Yellow
63	158	16	100	32	100	Green
159	208	16	100	32	100	Cyan
209	288	16	100	32	100	Blue
289	330	16	100	32	100	Magenta
331	341	16	100	32	100	Pink

Table 2. Summary of the features used to describe image content.

Group	Features	Components
Color	HSV Moments	9
	Skin Detector	1
Texture	NGTDM	5
	Wavelets	20
Edge Distribution	Canny Edge Statistics	11
	Edge Direction Histogram	18
Composisition	Fragmentation	1
	Symmetry	3

Four spatial composition features have been computed.[20] The first is fragmentation, which is defined as:

$$F = 1 - \frac{1}{N_r}\,, \tag{18}$$

where N_r is the number of color regions. High fragmentation indicates a complex image. The second feature refers to central dispersion and is measured as the average distance between the color regions positions and the center of the image, that is:

$$D = \frac{1}{N_r}\sum_{i=1}^{N_r} d(\underline{r}_i, \underline{c})\,, \tag{19}$$

where d denotes the Euclidean distance, $\underline{c} = (c_x, c_y)$ is the center of the image, and $\underline{r}_i = (r_{i,x}, r_{i,y})$ is the center of the ith color region.

The last two features refer to the horizontal and vertical symmetry of the image and are:

$$S_x = \sqrt[3]{\frac{1}{N_r}\sum_{i=1}^{N_r}(r_{i,x} - c_x)^3}\,, \tag{20}$$

$$S_y = \sqrt[3]{\frac{1}{N_r}\sum_{i=1}^{N_r}(r_{i,y} - c_y)^3}\,. \tag{21}$$

Altogether the features computed are 68, grouped as summarized in Table 2.

4. Experimental Results

The experiments were carried out inside a research project aimed to provide some semantic information to automatic image enhancement tools. The database we used was chosen to well represent the images that can be found on the web, or acquired by users using commercial equipment. We have tried to avoid having unwanted clusters of images (i.e. images derived from the same web site and/or with the same subject, for example horses), since that could somehow bias the classification experiments.

The database includes 9,000 photographs downloaded from the web, or acquired by a scanner or by digital cameras.

All the material varied in size, resolution and tonal depth. Before labeling and automatic indexing the images were resized to 128 pixels on the largest dimension, and proportionally on the other. Therefore image aspect ratio, and object aspect ratio, were maintained. The indoor class includes photographs of rooms, groups of persons, and details in which the context indicates that the photograph was taken inside. The outdoor class includes natural landscapes, buildings, city shots and details in which the context concurs to indicate that the photograph was taken outside. The close-up class includes portraits and photos of people and objects in which the context provides little or no information of where the photo was taken. Figures A.1–A.3 in the Appendix provide some examples for each class.

All the images have been labeled as indoor, outdoor or close-ups on the basis of the independent judgement of five observers. The agreement among them was rather high, but not complete, univocally labeling 78% of the images, while assigning different labels (such as indoor and close-up) to 22% of the images. Most of the confusion was between the close-up and the other categories (97% of the multiple labels). For each image the final label was assigned by majority vote. No images were excluded from the database, which composed of about 2,100 indoor images, 4,650 outdoor images and 2,250 close-ups.

4,500 images were randomly extracted from the database to form a training set, in which the three classes were equally represented. The remaining 4,500 images formed the test set on which we based the validation process. All the results presented here refer to this set.

In our experiments, we first built 10 trees using different subsets of the training set, and then used each tree to classify 4,500 photographs of the test set. Each subset contained 3,000 photographs randomly chosen from the training set in such a way that, once again, the three classes were equally represented. Each of the 10 trees was the most accurate tree within the sequence of subtrees obtained by applying the cost-complexity pruning described in Sec. 2. To select it, the accuracy of the subtrees was evaluated by classifying the 1,500 photographs not used for the training process. Table 3 shows the average result obtained using the ten single trees on the test set.

Table 3. Confusion matrix (average) obtained on the test set using single trees. Brackets contain minimum and maximun values.

		Predicted Class		
		Indoor	Outdoor	Close-up
True	Indoor	0.787 (0.762–0.821)	0.117	0.096
Class	Outdoor	0.085	0.817 (0.771–0.864)	0.098
	Close-up	0.079	0.077	0.844 (0.792–0.883)

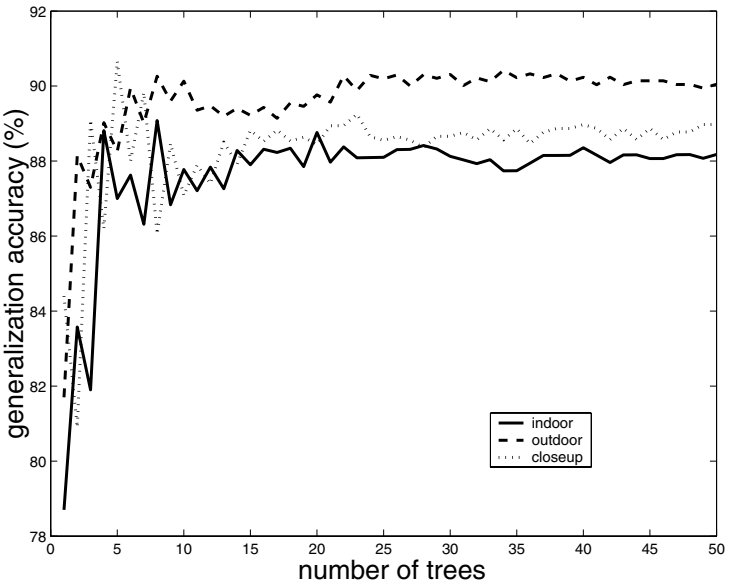


Fig. 2. Graph of the generalization accuracy versus the size of the forest for the three classes.

The subsequent step was the construction of 50 decision forests of increasing size. The forest of size k was obtained by making k bootstrap replicates of the training set and using them to build k trees. The classifications obtained with the different trees were then combined by means of a majority vote. As in the experiments with the single trees, each tree in each forest was the most accurate among the sequence of subtrees obtained by applying the pruning process. To do so each bootstrap replicate of the training set was subdivided in two subsets, one used to build the initial tree and perform the pruning, and the other to select the best tree in the sequence.

The results obtained by using the forests on the test set are shown in Fig. 2. As can be seen, for all three classes, forests of more than 20 trees achieve an accuracy that is nearly stable. Tables 4 and 5 show the results achieved by using the forests of 25 and 35 trees, respectively. The improvement over the results shown in Table 3, is significant: about 9% for the indoor class, 8% for the outdoor class and 5% for the close-up class.

Table 4. Confusion matrix obtained on the test set using the forest of 25 trees.

		Predicted Class		
		Indoor	Outdoor	Close-up
True	Indoor	0.881	0.050	0.069
Class	Outdoor	0.027	0.902	0.071
	Close-up	0.054	0.061	0.885

Table 5. Confusion matrix obtained on the test set using the forest of 35 trees.

		Predicted Class		
		Indoor	Outdoor	Close-up
True Class	Indoor	0.881	0.051	0.068
	Outdoor	0.028	0.903	0.069
	Close-up	0.052	0.061	0.887

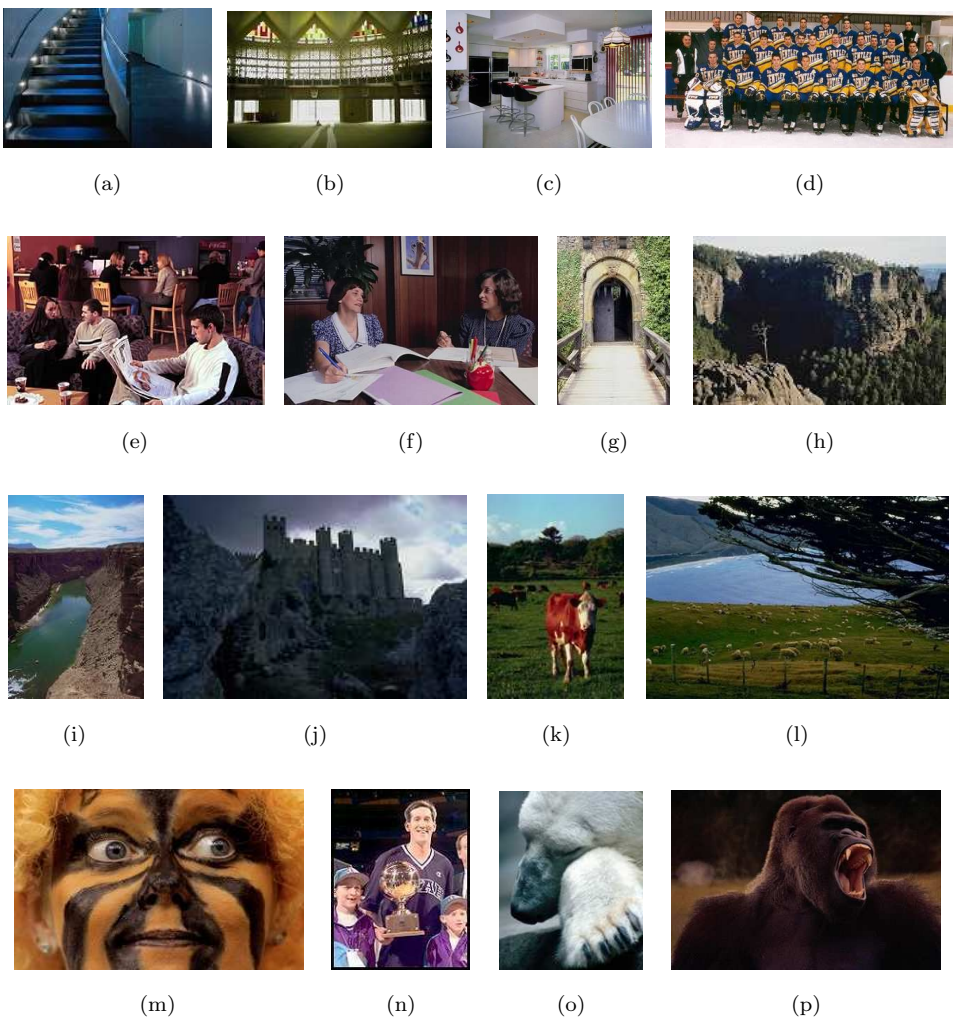


Fig. 3. Some misclassified images: (a–d) indoor images classified as outdoor; (e,f) indoor images classified as close-up; (g,h) outdoor images classified as indoor; (i–l) outdoor images classified as close-up; (m,n) close-up images classified as indoor; (o,p) close-up images classified as outdoor.

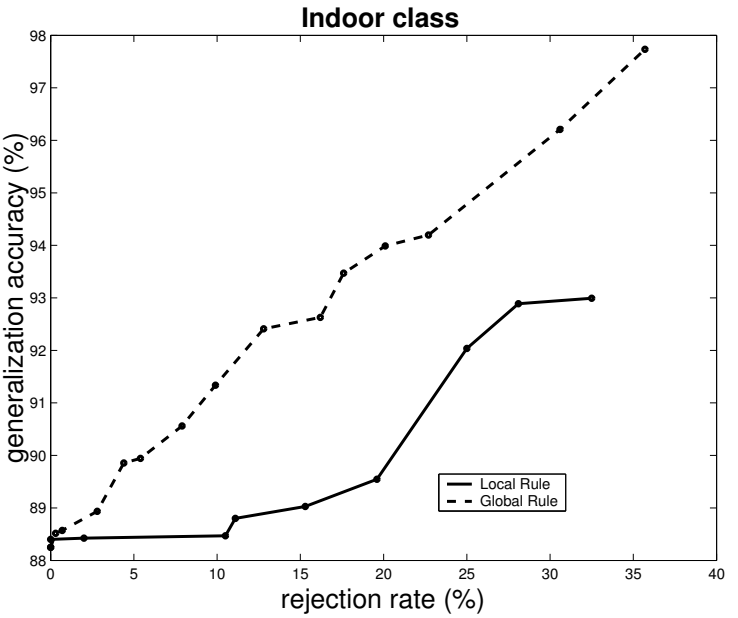


Fig. 4. Graph of the generalization accuracy versus the rejection rate, referred to the images not rejected.

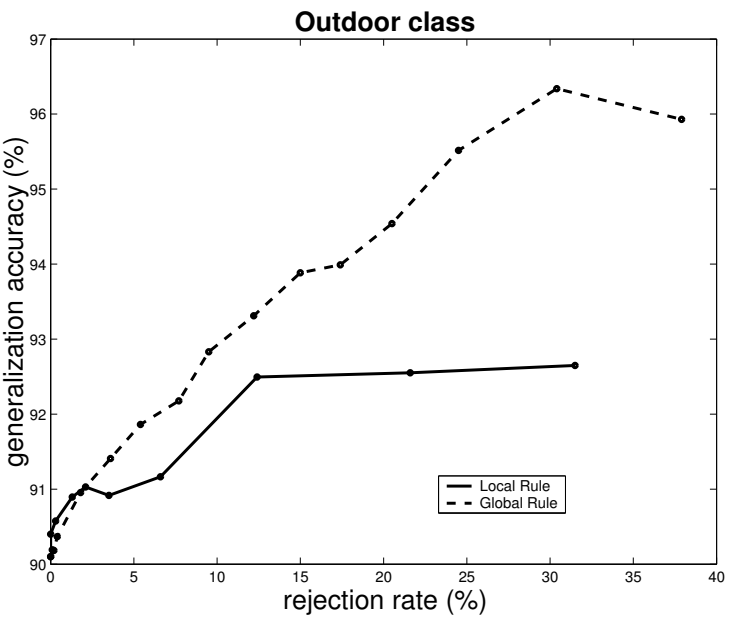


Fig. 5. Graph of the generalization accuracy versus the rejection rate, referred to the images not rejected.

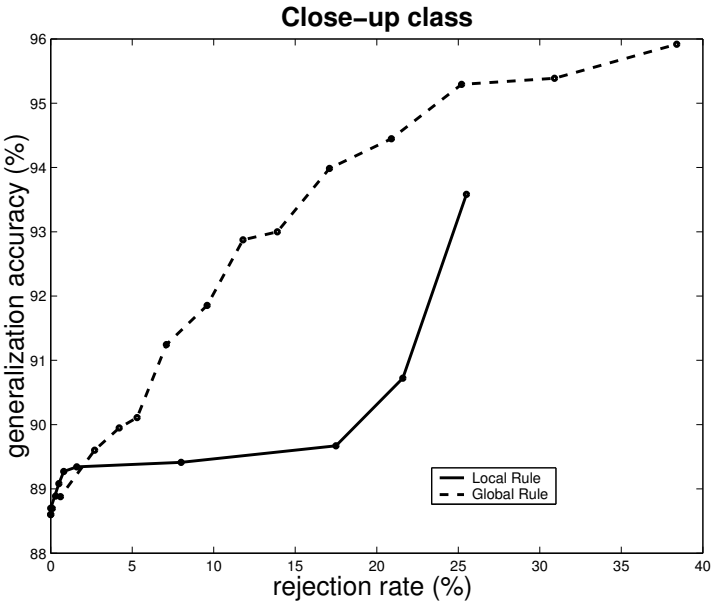


Fig. 6. Graph of the generalization accuracy versus the rejection rate, referred to the images not rejected.

The misclassified images are generally photographs that are either over- or under-exposed, or with a background that provides little information about the class to which the images belong. Indoor images misclassified as outdoor often show a window, while outdoor images misclassified as indoor are images of details of buildings, with little outdoor background. We may consider acceptable the misclassification of some close-up images as indoor or outdoor, and vice versa: it simply reflects the overlapping between the close-up class and the other categories. Figure 3 provides some examples of misclassified images.

In applying the rejection option, we faced two alternatives. One was to apply it to all or some of the forests, and then average the results; the other was to select one specific forest and apply the rejection option to that. To avoid a computational burden which seemed unjustified, we opted for the latter possibility and, on empirical grounds, selected the forest of 25 trees. The application of the rejection option to this forest produced the results presented here below.

Figures 4–6, for each class separately, show how the accuracy (referred to the remaining images) increases as the rejection rate increases, when either of the two rules used to implement the rejection option is applied. For all three classes the global rule outperforms the local rule, and with very satisfacory results.

Figures 7 and 8 show how the rejection rate for the three classes depends on the rejection threshold. We recall from Sec. 2 that for the global rule the rejection threshold is the percentage of trees in the forest producing the same classification of an image. For the local rule, instead, it is the resubstitution estimate of the accuracy within the terminal nodes.

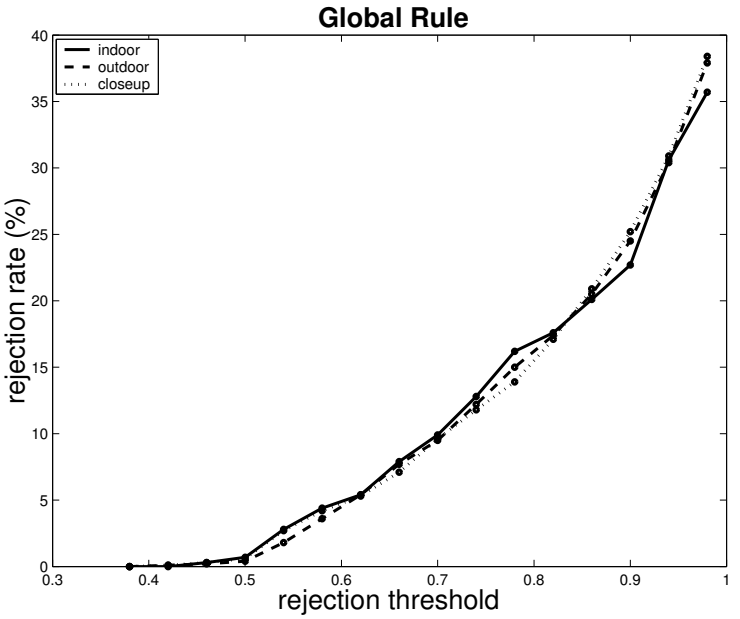


Fig. 7. Graph of the rejection rate versus the rejection threshold for the three classes when the global rule is applied.

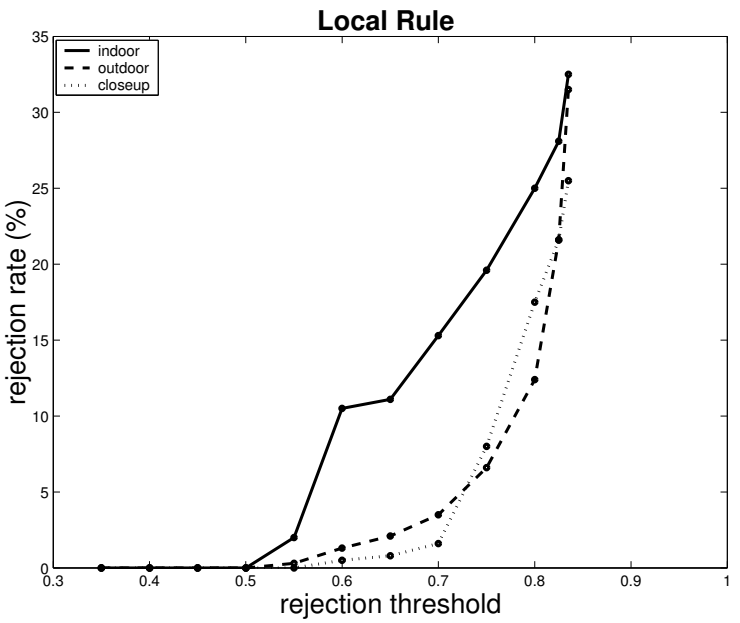


Fig. 8. Graph of the rejection rate versus the rejection threshold for the three classes when the local rule is applied.

Looking at Fig. 7, we see that when the global rule is applied, even when the threshold is set at its maximum value, the rejection rate is less than 40%. This means that about 60% of the images of the test set are classified with the high confidence that is derived from a nearly unanimous vote. We also note that when the threshold is set at a slightly lower value, as 0.8 for example, the rejection rate is less than 20%. If only a low threshold value is sufficient, such as, for example, 0.6, the rejection rate is very low, less than 5%. This holds for all three classes.

The first thing to be noted regarding the local rule is that the graph does not register the rejection rate depending on the rejection threshold for threshold values over than 0.8 (0.83 precisely). For these values the rejection rate of the indoor and outdoor classes is very high (nearly 95%), but this is not related to the unreliability of the classification results. To the contrary, it depends on the positive fact that most trees in the forest presented, for the indoor and outdoor classes, a limited number of large terminal nodes of accuracy between 0.8 and 0.9, and that many images of the test set ended up in these nodes. It is the rejection of these images which inflates the rejection rate. A large node with a high accuracy is most desirable: it means that many photographs of the class assigned to that node lie in a region of the feature space that the tree is able to detect. Moreover, the fact that many photographs of the test set end up in this node confirms that the set of conditions on the features defining the region occurs with a high frequency inside the class. Looking at Fig. 8, we also see that, when the local rule is applied, for rejection threshold values between 0.6 and 0.8, the rejection rate of the indoor class is at least 10% higher than that of the other classes. This signifies a greater difficulty in classifying indoor images.

The rejection of the images ending up in large terminal nodes of high accuracy could be prevented by incorporating in the local rule information about the size of the nodes. This would allow to deflate the influence of very small terminal nodes as

Table 6. Confusion matrix obtained on the test set using the forest of 25 trees and 10% rejection rate.

		Predicted Class		
		Indoor	Outdoor	Close-up
True	Indoor	0.913	0.035	0.052
Class	Outdoor	0.018	0.928	0.054
	Close-up	0.045	0.036	0.919

Table 7. Confusion matrix obtained on the test set using the forest of 25 trees and 30% rejection rate.

		Predicted Class		
		Indoor	Outdoor	Close-up
True	Indoor	0.962	0.017	0.021
Class	Outdoor	0.018	0.963	0.015
	Close-up	0.032	0.014	0.954

well, but it has the drawback of requiring the introduction of a further threshold. For this reason, and on account of the very good performance, and simplicity, of the global rule, we have not yet implemented the idea.

Tables 6 and 7 summarize the results obtained with 10% and 30% rejection rates. The information provided by the above tables combined with the information provided by Fig. 7, shows, for example, that the very high level of accuracy reached with 30% rejection rate corresponds to a rejection threshold of over 0.9. With 10% rejection rate the improvement in accuracy with respect to Table 4 is about 3% for the indoor class, 3% for the outdoor class, and 4% for the close-up class. With 30% rejection rate, the improvement increases to 8%, 6% and 7%, respectively.

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Fig. 9. Some rejected images.

Table 8. Summary of the results, expressed in terms of generalization accuracy.

	Indoor	Outdoor	Close-up
Single Tree	0.787	0.817	0.844
Decision Forest with 25 trees	0.881	0.902	0.885
10% rejection rate	0.913	0.928	0.919
30% rejection rate	0.962	0.963	0.954

Clearly, the rate that is acceptable is an application-dependent parameter. Table 8 summarizes in terms of generalization accuracy all the results presented above.

Figure 9 provides a few examples of rejected images.

The features which have been used more often in the construction of the forest of 25 are the skin detector, the vertical and horizontal edge directions, other statistics related to edge distribution, and dispersion. About half of the features have never been used.

5. Summary and Perspectives

The experimentation on indoor/outdoor/close-up classification described in this paper is part of a more general project concerning the design of intelligent digital cameras that can automatically and reliably adopt the most appropriate strategies of image enhancement, color processing and compression on the basis of the scene depicted. But the results could be also fruitfully applied in the framework of content-based retrieval: classification may significantly enhance retrieval systems by allowing semantically-adaptive searching and by filtering irrelevant classes during matching.

The experimental results we have presented here show that the use of decision forests of trees built according to CART methodology can provide very satisfactory results in dealing with the difficult problem of identifying semantic classes of photographs, such as indoor, outdoor and close-up, simply on the basis of low-level features related to color, texture and composition of the images. The accuracies we reported refer to the test set alone and are not averaged on the entire database, which includes the training set.

The results also show that accuracy can be notably improved by using a rejection option. In image processing a rejection scheme can prevent the production of artifacts that may be derived from activating unappropriate processing algorithms.[11] In the retrieval framework, as Vailaya points out,[33] it can be applied to automatically generate reliable high-level indices in large image databases. Setting a high rejection rate can help in the classification of only those images in which the classifier has a high confidence. For instance, for each of the three classes taken into account, we achieved an accuracy of over 95% at the 30% rejection rate; for databases of milions of images, the automatic and reliable classification of 70% of the images would considerably reduce the manual effort.

We decided to experiment with CART trees for several reasons. We have already mentioned that they make powerful use of conditional information and high

dimensionality, and perform automatic dimensionality reduction. Noisy features have no detrimental effect on their performance, as they do, instead, on that of nearest neighbor rules.[19] Moreover, CART trees do not require assumptions about the probability distribution of the features. They provide not only a classification rule, but also a clear characterization of the conditions that drive the classification, and allow us to assign a level of confidence in classification results as well. Finally, they are robust with respect to outliers.

Other properties, not exploited in the present study, might be very useful in future experimentation related to the design of image filters for the WEB. Qualitative features, such as image format, could be dealt with as easily as quantitative features, and different misclassification costs could be taken into account. This would be important in dealing with pornographic images. Moreover, since CART methodology allows for the presence of missing values both in the training set and in new cases to be classified, in image classification we could also take advantage of ancillary information, such as image captions, which may not always be available.

We believe that the performance will scale well to different and, perhaps, larger databases of photographs. However, a number of issues deserve future work. As far as image description is concerned, we must design a more powerful skin detector, since we have observed that classification results were worst when significant parts of the images were occupied by skin regions. As for the classification process, we may decide to refine the local rejection rule by incorporating in it some information about the size of the terminal nodes, as already observed in the discussion of the results. Moreover, we need to develop some tool that allows us to gain an insight into the role the different features play in the construction of the forest, since forests are not as highly interpretable as single trees. Such a tool would have to summarize the contribution that each feature provides, inside the forest, in separating one class from the others, and also help to detect possible masking effects among the features. Features which are little used could indeed be masked by others. A thorough analysis of the role of the features in the classification process will help in the re-examination of the features set that is necessary at this stage of our project. This review process could also benefit very much from the use of unsupervised learning techniques, such as clustering, which are indicated as a promising research issue for filling the gap between high-level semantic concepts and low-level features in multimedia applications.[21,26] Middle or high level features could be considered as well.[20,26]

Finally, since the problem we face is such that some images fall undeniably into several classes simultaneously, a fuzzy interpretation of the classification results provided by the forest could be investigated.

Acknowledgments

This investigation was performed as a part of a ST Microelectronics research contract. The authors thank ST Microelectronics for permission to present this paper. They thank also the referees for the very useful comments.

Appendix

The following figures show examples of photographs of the database, separately for each of the three classes.

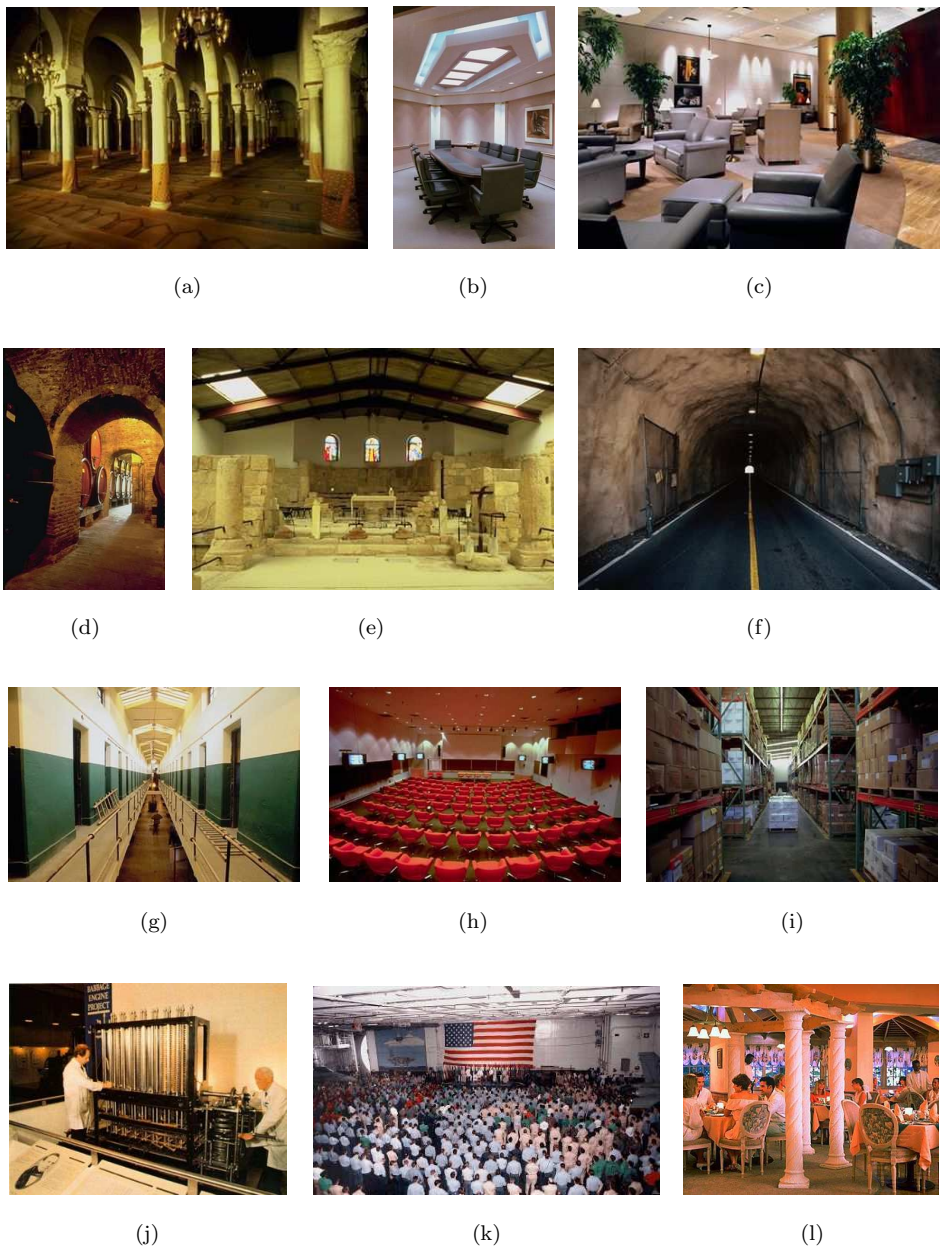


Fig. A.1. Examples of indoor photographs.

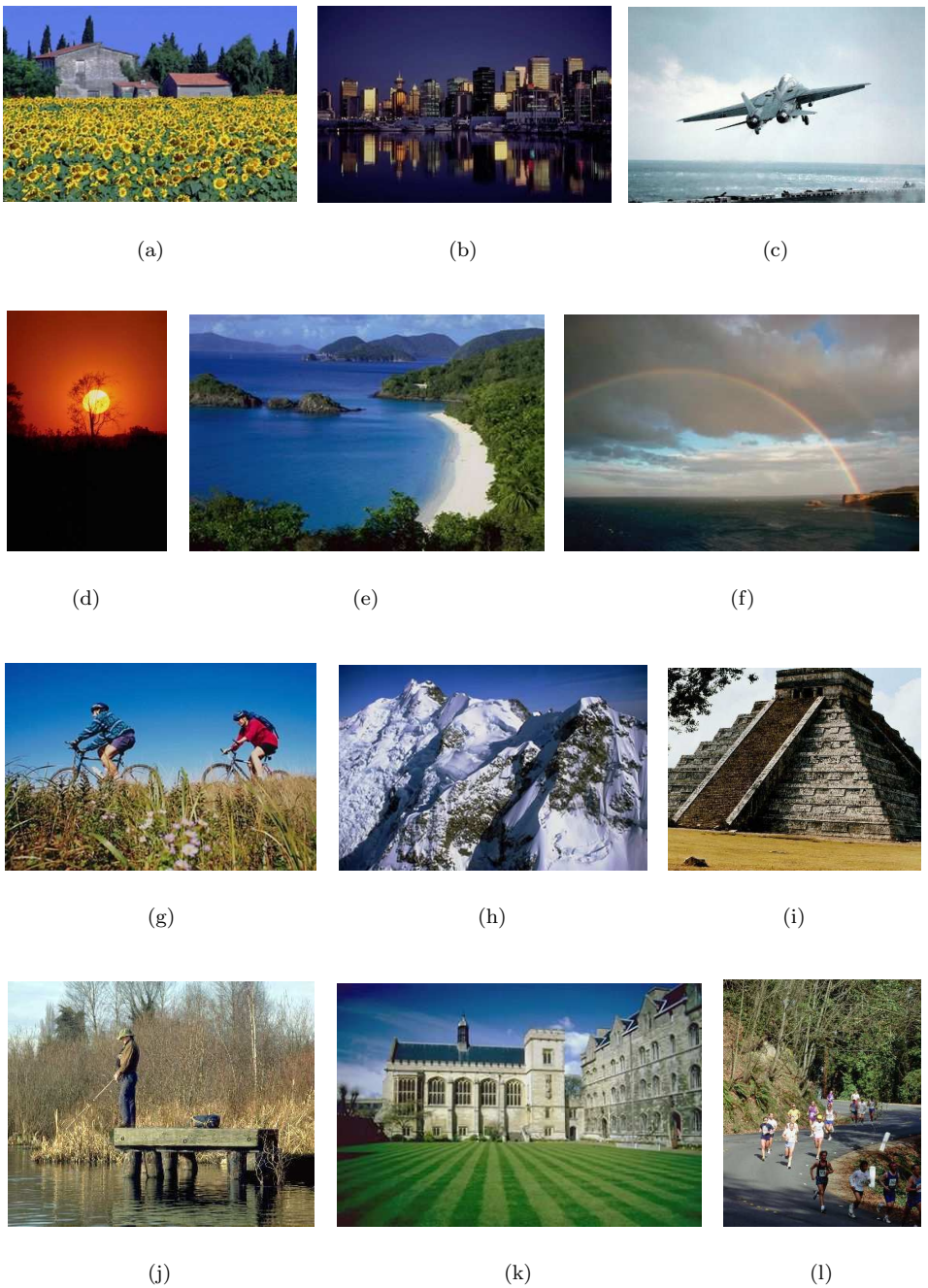
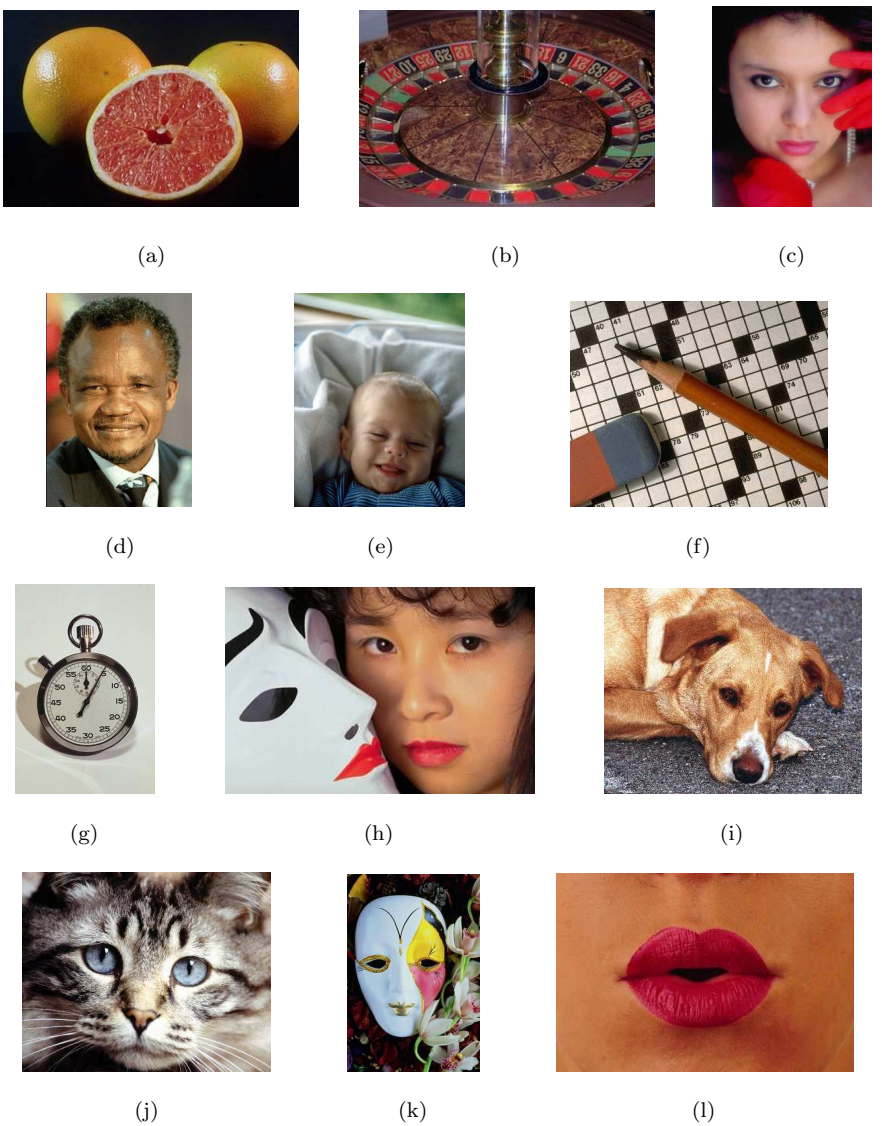


Fig. A.2. Examples of outdoor photographs.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Fig. A.3. Examples of close-up photographs.

References

1. M. Amadasun and R. King, Textural features corresponding to textural properties, *IEEE Trans. Syst. Man Cybern.* **19**(5) (1989) 1264–1274.
2. V. Athitsos and M. Swain, Distinguishing photographs and graphics on the World Wide Web, in *Proc. Workshop in Content-based Access to Image and Video Libraries*, 1997, pp. 10–17.
3. L. Breiman, J. H. Friedman, R. A. Olshen and C. J. Stone, *Classification and Regression Trees* (Wadsworth and Brooks/Cole, 1984).
4. L. Breiman, Bagging predictors, *Mach. Learn.* **24** (1996) 123–140.

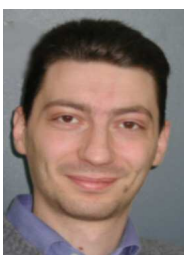
5. P. Bühlmann and B. Yu, Analyzing bagging, *Ann. Statist.* **30** (2002) 927–961.
6. J. Canny, A computational approach to edge detection, *IEEE Trans. Patt. Anal. Mach. Intell.* **8** (1986) 679–698.
7. P. Ciocca and R. Schettini, A relevance feedback mechanism for content-based retrieval, *Inform. Process. Manag.* **35** (1999) 605–632.
8. T. G. Dietterich, Ensamble methods in machine learning, in *Proc. First Int. Workshop on Multiple Classifier Systems*, Lecture Notes in Computer Science (Springer, 2000), pp. 1–15.
9. S. Fischer and H. Bunke, Automatic identification of diatoms using decision forests, in *Proc. Second International Workshop on Machine Learning and Data Mining in Pattern Recognition*, Lecture Notes in Artificial Intelligence (Springer, 2001), pp. 173–183.
10. I. Gagliardi and R. Schettini, A method for the automatic indexing of color images for effective image retrieval, *New Rev. Hyperm. Multim.* **3** (1997) 201–224.
11. F. Gasparini, R. Schettini and P. Gallina, Innovative algorithm for cast detection, in *Proc. SPIE Internet Imaging III*, Vol. 4672, 2002, pp. 280–286.
12. T. Gevers and A. W. M. Smeulders, PicTo Seek: combining color and shape invariant features for image retrieval, *IEEE Trans. Imag. Process.* **19**(1) (2000) 102–120.
13. D. Hand, *Construction and Assessment of Classification Rules* (Wiley, 1997).
14. T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning* (Springer, 2001).
15. T. K. Ho, The random subspace method for constructing decision forests, *IEEE Trans. Patt. Anal. Mach. Intell.* **20** (1998) 832–844.
16. F. Idris and S. Panchanathan, Storage and retrieval of compressed images using wavelet vector quantization, *J. Vis. Lang. Comput.* **8** (1997) 289–301.
17. A. Jain, R. Duin and J. Mao, Statistical pattern recognition: a review, *IEEE Trans. Patt. Anal. Mach. Intell.* **22**(1) (2000) 4–37.
18. R. Lienhart and A. Hartmann, Classifying images on the WEB automatically, *J. Electron. Imag.* **11**(4) (2002) 445–454.
19. G. J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition* (Wiley, 1992).
20. Y. Miyake, H. Saitoh, H. Yaguchi and N. Tsukada, Facial pattern detection and color correction from television picture for newspaper printing, *J. Imag. Technol.* **16** (1990) 165–169.
21. Y. Rui, T. S. Huang and S. F. Chang, Digital image/video library and MPEG-7: standardization and research issues, in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 1998, pp. 3785–3788.
22. R. Schettini, C. Brambilla, G. Ciocca, A. Valsasna and M. De Ponti, A hierachical classification strategy for digital documents, *Patt. Recogn.* **35** (2002) 1759–1769.
23. R. Schettini, C. Brambilla, A. Valsasna and M. De Ponti, Content-based classification of digital documents, in *Proc. First Int. Workshop on Pattern Recognition in Information System*, 2001, pp. 161–170.
24. R. Schettini, G. Ciocca and S. Zuffi, Indexing and retrieval in colour image databases, *Color Imaging Science*: *Exploiting Digital Media* (Wiley, 2001), pp. 183–211.
25. P. Scheunders, S. Livens, G. Van de Wouwer, P. Vautrot and D. Van Dyck, Wavelet-based texture analysis, wcc.ruca.ua.ac.be/˜livens/WTA/, 1997.
26. A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain, Content-based image retrieval at the end of the early years, *IEEE Trans. Patt. Anal. Mach. Intell.* **22**(12) (2000) 1349–1380.
27. M. Stricker and M. Orengo, Similarity of color images, in *Proc. SPIE Storage and*

Retrieval for Image and Video Databases III Conf., 1995, pp. 381–392.
28. M. Stricker and A. Dimai, Color indexing with weak spatial constraints, in *Proc. Storage and Retrieval for Image and Video Databases IV Conf.*, 1996, pp. 29–40.
29. M. Stricker and A. Dimai, Spectral covariance and fuzzy regions for image indexing, *Mach. Vis. Appl.* **10** (1997) 66–73.
30. M. Szummer and R. Picard, Indoor-outdoor image classification, in *Proc. Int. Workshop on Content-Based Access of Image and Video databases*, 1998, pp. 42–51.
31. A. Vailaya, M. Figueiredo, A. K. Jain and H.-J. Zhang, Image classification for content-based indexing, *IEEE Trans. Imag. Process.* **10**(1) (2001) 117–130.
32. A. Vailaya, A. K. Jain and H.-J. Zhang, On image classification: city images versus landscapes, *Patt. Recogn.* **31** (1998) 1921–1936.
33. A. Vailaya and A. Jain, Reject option for VQ-based Bayesian classification, *15th Int. Conf. Pattern Recognition*, Barcelona, Spain, 2000.
34. E. Yiu, *Image Classification Using Color Cues and Texture Orientation*, Department of Electrical Engineering and Computer Science, MIT, Master thesis, 1996.

Raimondo Schettini is an Associate Professor at DISCo (Dipartimento di Informatica, Sistemistica e Comunicazione), University of Milano Bicocca, where he is in charge of the Imaging and Vision Lab. He has been associated with Italian National Research Council (CNR) since 1987. He has been a team leader in several research projects and published more than 140 refereed papers on image processing, analysis and reproduction, and on image content-based indexing and retrieval. He has been General Co-Chairman of the 1st Workshop on Image and Video Content-based Retrieval (1998), of the First European Conference on Color in Graphics, Imaging and Vision (CGIV'2002), and of the EI Internet Imaging Conferences (2000–2004). He has been guest editor of two special issues about Internet Imaging (*J. Electronic Imaging*, 2002), and Color Image Processing and Analysis (*Pattern Recognition Letters*, 2003), and he is currently associate editor of the *Pattern Recognition Journal*.

Carla Brambilla is a senior researcher at the Institute of Applied Mathematics and Information Technologies of the Italian National Research Council.

Her research focuses on classification and regression trees, generalized linear models and extensions, and survival analysis.



Claudio Cusano is a Ph.D. student at DISCo, (Dipartimento di Informatica, Sistemistica e Comunicazione), University of Milano-Bicocca, where he received his degree in computer science. Since April 2001, he has been a fellow of the Imaging and Vision Laboratory at the ITC Institute of the Italian National Research Council.

His current research activity focuses on image analysis and classification, and on tridimensional imaging.

Gianluigi Ciocca received his degree in computer science at the University of Milan in 1998, and since then he has been a fellow at the Institute of Multimedia Information Technologies of the Italian National Research Council, where his research has focused on the development of systems for the management of image and video databases and the development of new methodologies and algorithms for automatic indexing. He is currently a Ph.D. student in computer science at DISCo (Dipartimento di Informatica, Sistemistica e Comunicazione), University of Milano–Bicocca, working on video analysis and abstraction.