Color Image Classification Using Tree Classifiers

R. Schettini°, C. Brambilla, G. Ciocca°, M. De Ponti[§]*
ITIM°, IAMI, Consiglio Nazionale delle Ricerche*
[§]STMicroelectronics, TPA Group, Printer Division

Abstract

The paper addresses the problem of how to efficiently and effectively classify color images in predefined classes purely on the basis of low-level feature analysis using tree classifiers. The method proposed here has been tested on the specific high-level classification problem of distinguishing photographs from artworks. Preliminary results are reported.

Introduction

Content-based image classification has emerged an important area in multimedia computing due to the rapid development of digital imaging, storage and networking technologies.[1] We believe that content-based analysis and classification can also be fruitfully applied in cross-media color reproduction. Recognizing the class to which a processed image is likely to belong would allow the CMS to process the image according to specific strategies, perform color adjustments, or obtain a more pleasant (or preferred) color reproduction without requiring user interaction. We have addressed here the problem of how to efficiently and effectively classify color images in predefined classes purely on the basis of low-level feature analysis. The proposed method has been tested on the specific high-level classification problem of distinguishing photographs from artworks.

Tree Classifiers

There are many methods of classification and it is well known that works well in one case may be not satisfactory in another. We decided to experiment with tree classifiers since they allow us to handle the mixture of features types and the co-existence of different relationships between the features in different regions of the features space in a very natural way. Moreover, they provide a clear characterization of the conditions that drive the classification, that is, of the conditions that determine when an image belongs to one class rather than to another. Last, but not least, they are very easy to use. Although the construction of trees dates back to the sixties, where they were first employed in the social sciences,[9] it was the work by Breiman et al.[3] in the eighties to have a seminal influence both in bringing tree methodology to the attention of the scientific community and in stimulating the development of new strategies and algorithms. Today trees are used in a great variety of applications in fields ranging from medicine to meteorology, and from marketing to chemistry. Many and up-to-date references to their use are given in [11]. An evaluation of the tree approach to classification on several databases, together with a comparison with other approaches can be found in [8].

Broadly speaking, tree classifiers are trees constructed by recursively partitioning the predictor space, each split being formed by the conditions related to the predictor values. The construction process is based on training sets consisting of cases whose class is known. In our problem the predictors are the features indexing the images and the training sets are sets of images whose semantic class is known. Once the tree has been constructed, a class is assigned to each of the terminal nodes. This is what actually makes the tree a classifier: when a new case is processed by the tree, its predicted class is given by the class attached to the terminal node into which the case finally moves on the basis of its predictor values. The classes are assigned to the terminal nodes in such a way that, if a case falls into the node, the probability of being misclassified, or, more generally, the expected cost of the misclassification is minimized.

The splitting process must essentially solve two problems: find the candidate splits, and define the goodness of a split. The candidate splits are generated by a set of questions on the values of the predictors, which are different depending on the nature of the predictors themselves. For a numerical predictor, for example, the admissible questions are: $\{is\ x < c?\}$, where x denotes the value of the predictor and c ranges over the real line. At each step of the process, all the predictors are searched one by one and, for each predictor, the best split (in the sense defined below) is found. Then the best single predictor splits are compared, and the best of these selected. The process starts at the root and continues on until some stopping rule is satisfied. This usually produces a large tree, which can be pruned in order to generate a reasonable number of sub-trees decreasing in size among which the tree to use as classifier can then be chosen. This tree is the one for which the total probability of misclassification or, more generally, the total expected cost of misclassification, evaluated for a test set, or by cross-validation, is smallest. The pruning process is strongly advisable when the cost of having a large tree is not justified by a significant improvement in terms of performance.

As regards the goodness of the splits, the central idea proposed in [3] is to select the splits so that the data in the descendant nodes are purer than the data in the original ones. To do so, different functions of the impurity of the nodes are introduced, and the decrease in the value of the choosen function produced by a split is taken as a measure of the goodness of the split itself. Another possibility, suggested in [5], is to evaluate the goodness of the splits by using the reduction in deviance, where the deviance is a function of the likelihood, signifying the discrepancy of a fit with the data.[7] This approach views a tree as providing a model whose parameters are the probabilities of a multinomial distribution.

Image Description

Obviously the significance of the training set and the quality of the features used to described the image content are essential factors for a good classification. We have systematically studied how to extract low-level representations (in terms of color, texture, and shape features) from the images describing their pictorial content, taking into account three basic properties:[4,6]

- perceptual similarity (the feature distance between two images is large only if the images are not "similar"),
- efficiency (the features can be rapidly computed) and
- economy (their dimensions are small in order not to affect retrieval efficiency).

The features listed below constitute a general purpose library of low-level features that can be calculated on the global image and/or on sub-images, obtained by dividing the original image in different ways and then eventually re-combined:

- the Color Coherence Vectors (CCV)[10] in the CIELAB color space quantized in 64 colors. CVV buckets color pixels as coherent or incoherent according to whether or not they belong to a large-similarly colored regions. Before CCV computation the image is blurred by local averaging in a 3x3 neighbour.
- a histogram of the transition in color (a CIELAB color space quantized in 11 colors, namely red, orange, yellow, green, blue, purple, pink, brown, black, grey and white);[6]
- the moments of inertia of the distribution of colors in the unquantized CIELAB color space;[13]
- a histogram of opportunely filtered contour directions (only high gradient pixels are considered) Edges are extracted by Canny's edge detectors, and the corresponding edge directions quantized in 72 bins at 5° intervals. To compensate for different image sizes, the histograms are normalized with respect to the total number of edge pixels detected in the image;[4]
- the mean and variance of the absolute values of the coefficients of the sub-images of the first three levels of the multi-resolution Daubechies wavelet transform of the luminance image (See Figure 2); [12]
- the estimation of statistical features based on the Neighborhood Gray-Tone Difference Matrix (NGTDM), i.e. coarseness, contrast, busyness, complexity, and strength. These features are computed as proposed by Amadasum and King;[2]
- the spatial composition of the color regions identified by the process of quantization in 11 colors;[4] i) fragmentation (the number of color regions), ii) distribution of the color regions with respect to the center of the image; iii) distribution of the color regions with respect to the x axis, and with respect to the y axis.

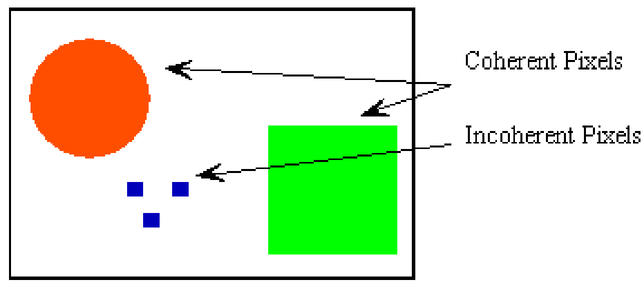

Figure 1. CVV. Example of coherent or incoherent pixels.

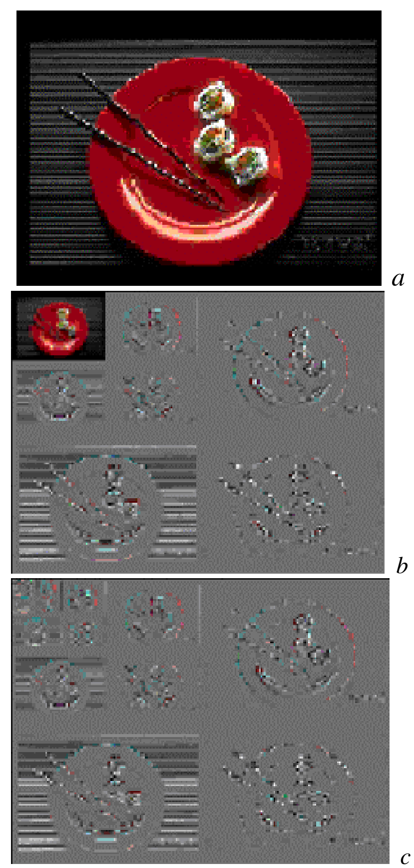

Figure 2. a) original image, b) two-steps multiresolution wavelet transform, c) multiresolution wavelet transform

We have used this somewhat redundant image description as none of the features taken singly identifies an image univocally: completely different images may yield similar feature values. The very different natures of the indices selected should limit the risk of having different images correspond to very close points in the feature space.

Preliminary Results

Many specific image classes that would require ad-hoc processing. For the moment we have experimented our approach on the specific high-level classification problem of classifying a given image either as photograph or an artwork. Of course both the classes could be further split using the same classification scheme.

We have employed several training and test sets composed of several hundred images, up to 1500 for the training sets, and 1250 images for the test sets. The subject matter of the images was rather heterogeneous; however, no mixed images, such as photographs with overlaid texts, or framed, were used. The image labels were assigned, and independently verified by two of the authors. Classification accuracy was, on average, about 95% on the training sets, and about 91% on the test sets.

Figure 3. Examples of misclassified images: photographs classified as artwork.

To achieve these results the classifier exploited only about ten percent of the features computed, consequently only these could have been computed on the test set.

Figures 3 and 4 show a representative set of misclassified images.

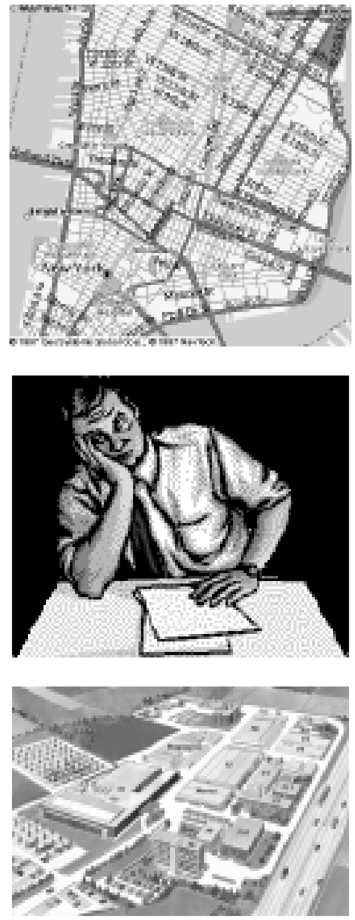

Figure 4. Examples of misclassified images: artworks classified as photographs.

Conclusions

We are currently testing our approach on additional classes, enlarging the feature library to include other, more specific features for discriminating photographs, and attempting to find a way to cope with the weakest point of the chosen classifier, the fact that it can not reject images that do not belong to any of the predefined classes.

Our final goal is to compact a n-class classifier into a single hierarchical classifier.

Acknowledgement

This investigation was performed as a part of a ST Microelectronics' research contract. The authors thank ST Microelectronics for permission to present this paper.

References

1. Aigrain O., Zhang H. & Petkovic D. (1996) Content-Based Representation and Retrieval of Visual Media: A State-of-the-Art Review. *Multimedia Tools and Applications*, 3,179-182.
2. Amadasun M. & King R. (1989) Textural features corresponding to textural properties, *IEEE Transaction on System, Man and Cybernetics,* 19(5), 1264-1274.
3. Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, C.J. (1984) Classification and regression trees, Wadsworth and Brooks/Cole.
4. Ciocca G., R. Schettini (1999) A relevance feedback mechanism for content-based image retrieval, *Information Processing and Management* (in print).
5. Clark, L.A. and Pregibon, D. (1992) Tree-based models. In Statistical models in S, eds. Chambers, J.M. and Hastie, T.J., 377-420, Chapman and Hall.
6. Gagliardi I. and Schettini R. (1997) A method for the automatic indexing of color images for effective image retrieval. *The New Review of Hypermedia and Multimedia*, 3, 201-224.
7. McCullagh, P. and Nelder, J.A., (1989) Generalized linear models, Chapman and Hall.
8. Michie, D. Spiegelhalter, D.J. and Taylor C.C. (eds.) (1994), Machine learning, neural and statistical classification, Ellis Horwood.
9. Morgan J.N. and Sonquist, J.A. (1963) Problems in the analysis of survey data, and a proposal. *Journal of the American Statistical Association*, 58, pp. 415-434.
10. Pass G., Zabih R. and Miller J. (1996) Comparing Images Using Color Coherence Vectors. *Fourth ACM Multimedia 96 Conference.*
11. Steinberg, D. and Colla, P. (1997), CART, Salford Systems.
12. Scheunders P., Livens S., Van de Wouwer G., Vautrot P., Van Dyck D. (1997) Wavelet-based Texture Analysis. *International Journal Computer Science and Information management,* (wcc.ruca.ua.ac.be/~livens/WTA/).
13. Stricker M.A and Orengo M. (1995) Similarity of Color Images. *SPIE Storage and Retrieval for Image and Video Databases III Conference.*