Object Class Recognition using Images of Abstract Regions [*]

Yi Li†, Jeff A. Bilmes‡, and Linda G. Shapiro†‡
† Department of Computer Science and Engineering
‡ Department of Electrical Engineering
University of Washington
Seattle, WA 98195-2350
{shapiro@cs,yi@cs,bilmes@ee}.washington.edu

Abstract

With the advent of many large image databases, both commercial and personal, content-based image retrieval has become an important research area. While most early efforts retrieved images based on appearance, it is now recognized that most users want to retrieve images based on the objects present in them. This paper addresses the challenging task of recognizing common objects in color photographic images. We represent images as sets of feature vectors of multiple types of abstract regions, which come from various segmentation processes. We model each abstract region as a mixture of Gaussian distributions over its feature space. We have developed a new semi-supervised version of the EM algorithm for learning the distributions of the object classes. We use supervisory information to tell the procedure the set of objects that exist in each training image, but we do not use any such supervisory information about where (ie. in which regions) the objects are located in the images. Instead, we rely on our EM-like algorithm to break the symmetry in an initial solution that is estimated with error. Experiments are conducted on a set of 860 images to show the efficacy of our approach.

1. Introduction

Recognizing classes of objects in ordinary color photographic images is a difficult and challenging problem. Images may contain many different common objects, from different viewpoints, and in different arrangements. In this scenario, alignment-based techniques [4] are not appropriate, since they are intended for recognizing particular objects, and appearance-based techniques that attempt classification of the entire image [7][10] are also not suitable. Region-based techniques [1][9][3] require presegmentation of the image into regions of interest. In most applications, the reliability of image segmentation techniques has been a problem for object recognition, but newer image segmentation algorithms [6][8] that use both color and texture can now partition an image into regions that can, in many cases, be identified as classes of natural objects. Furthermore, a new mid-level feature called a *consistent line cluster* [5] can produce regions that often correspond to man-made structures. Since regions used in recognition can come from several different segmentation processes, we will refer to the regions we use in our work as *abstract regions*.

The idea behind the abstract region approach is that all features are image regions, each with its own set of attributes. The regions we have used to start our work are color regions and texture regions. We intend to add other types of abstract regions, including structure regions, axes of symmetry and major vertical line segments, in later work. Another possibility for abstract regions are the square patches selected by an entropy-based feature detector [11] that were successfully used in a new and promising approach to object class recognition [2] that models classes as flexible configurations of parts.

We have developed a new method for object recognition that uses whole images of abstract regions, rather than single regions for classification. A key part of our approach is that we do not need to know where in each image the objects lie. We only utilize the fact that objects exist in an image, not where they are located. We have designed an extended EM-like procedure that begins by computing an average feature vector over all regions in all images that contain a particular object. It relies on the fact that such an average feature vector is likely to retain attributes of the par-

[*] This research was supported by the National Science Foundation under Grants IRI-9711771, IIS-0097329, and IIS-0093430. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the National Science Foundation.

ticular object, even though the average contains instances of regions that do not contribute to that object. From these initial estimates, which are full of errors, the procedure iteratively re-estimates the parameters to be learned. It is thus able to compute the probability that object o is in image I given the set of feature vectors for all the regions of I. This paper describes our approach and illustrates its use with color and texture regions. In Section 2 we formalize our approach, in Section 3 we describe our experiments and results, and in Section 4 we discuss the implications of the results.

2. Methodology

We are given a set of training images, each containing one or more object classes, such as grass, trees, sky, houses, zebras, and so on. Each training image comes with a list of the object classes that can be seen in that image. There is no indication of where the objects appear in the images. We would like to develop classifiers that can train on the features of the abstract regions extracted from these images and learn to determine if a given class of object is present in an image.

Let T be the set of training images and O be a set of m object classes. Suppose that we have a particular type a of abstract region (e.g. color) and that this type of region has a set of n^a attributes (e.g. (H,S,I)) which have numeric values. Then any instance of region type a can be represented by a feature vector of values $r^a = (v_1, v_2, \ldots, v_{n^a})$. Each image I is represented by a set F_I^a of type a region feature vectors. Furthermore, associated with each training image $I \in T$ is a set of object labels O_I, which gives the name of each object present in I. Finally, associated with each object o is the set $R_o^a = \bigcup_{I: o \in O_I} F_I^a$, the set of all type a regions in training images that contain object class o.

Our approach assumes that each image is a set of regions, each of which can be modeled as a mixture of multivariate Gaussian distributions. We assume that the feature distribution of each object o within a region is a Gaussian $N_o(\mu_o, \Sigma_o)$, $o \in O$ and that the region feature distribution is a mixture of these Gaussians. We have developed a variant of the EM algorithm to estimate the parameters of the Gaussians. Our variant is interesting for several reasons. First, we keep fixed the component responsibilities to the object priors computed over all images. Secondly, when estimating the parameters of the Gaussian mixture for a region, we utilize only the list of objects that are present in an image. We have no information on the correspondence between image regions and object classes. The vector of parameters to be learned is:

$$\lambda = (\mu_{o1}^a, \ldots, \mu_{om}^a, \mu_{bg}^a, \Sigma_{o1}^a, \ldots, \Sigma_{om}^a, \Sigma_{bg}^a)$$

where $\{\mu_{oi}^a, \Sigma_{oi}^a\}$ are the parameters of the Gaussian for the ith object class and $\{\mu_{bg}^a, \Sigma_{bg}^a\}$ are the parameters of an additional Gaussian for the background. The purpose of the extra model is to absorb the features of regions that do not fit well into any of the object models, instead of allowing them to contribute to, and thus bias, the true object models. The label bg is added to the set O_I of object labels of each training image I and is thus treated just like the other labels.

The initialization step, rather than assigning random values to the parameters, uses the label sets of the training images. For object class $o \in O$ and feature type a, the initial values are

$$\mu_o^a = \frac{\sum_{r^a \in R_o^a} r^a}{|R_o^a|} \tag{1}$$

$$\Sigma_o^a = \frac{\sum_{r^a \in R_o^a} [r^a - \mu_o^a][r^a - \mu_o^a]^T}{|R_o^a|} \tag{2}$$

Note that the initial means and covariance matrices most certainly have errors. For example, the Gaussian mean for an object in a region is composed of the average feature vector over all regions in all images that contain that object. This property will allow subsequent iterations by EM to move the parameters closer to where they should be. Moreover, by having each mean close to its true object, each such subsequent iteration should reduce the strength of the errors assigned to each parameter.

In the E-step of the EM algorithm, we calculate:

$$\begin{gathered} p(r^a | o, \mu_o^a(t), \Sigma_o^a(t)) = 0, \text{ if } o \notin O_I, \text{else} \\ \frac{1}{\sqrt{(2\pi)^{n^a} |\Sigma_o^a(t)|}} e^{-\frac{1}{2}(r^a - \mu_o^a(t))^T (\Sigma_o^a(t))^{-1} (r^a - \mu_o^a(t))} \end{gathered} \tag{3}$$

$$p(o | r^a, \lambda(t)) = \frac{p(r^a | o, \mu_o^a(t), \Sigma_o^a(t)) p(o)}{\sum_{j \in O_I} p(r^a | j, \mu_j^a(t), \Sigma_j^a(t)) p(j)} \tag{4}$$

where $p(o) = \frac{|\{I | o \in O_I\}|}{|T|}$. Note that when calculating $p(r^a | o, \mu_o^a(t), \Sigma_o^a(t))$ in (3) for region vector r^a of image I and object class o and when normalizing in (4), we use only the set of object classes of O_I, which are known to be present in I. The M-step follows the usual EM process of updating μ_o^a and Σ_o^a. After multiple iterations of the EM-like algorithm, we have the final values μ_o^a and Σ_o^a for each object class o and the final probability $p(o|r^a)$ for each object class o and feature vector r^a. Now, given a test image I we can calculate the probability of object class o being in image I given *all* the region vectors r^a in I:

$$p(o | F_I^a) = \max_{r_a \in F_I^a} p(o | r^a) \tag{5}$$

We use max instead of sum, because each image has a different number of regions, and summing will favor classes with multiple regions in the same image. This is still for a single type of abstract region a. We will describe two methods to handle multiple types of abstract regions in our experiments.

3. Experiments and Results

Our color regions are produced by a two-step procedure: 1) A K-means variant performs clustering in HSI space. 2) Tiny regions are merged into similarly-colored adjacent larger ones. Our texture regions come from a color-guided texture segmentation process. Color segmentation is first performed, and then pairs of regions are merged if after a dilation they overlap by more than 50%. Each of the merged regions is segmented using the K-means variant algorithm on the Gabor texture coefficients. Figure 1 illustrates the color and texture regions for two representative images.

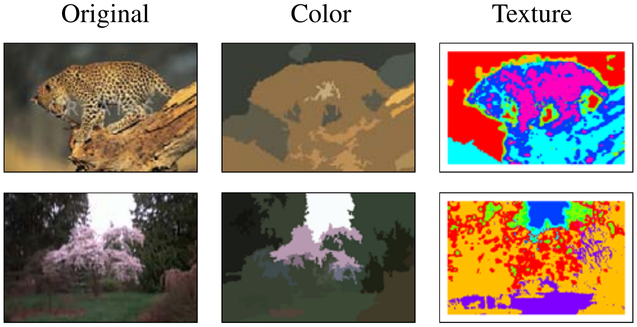

Figure 1. The abstract regions constructed from a set of representative images using color clustering and color-guided texture clustering.

Since our abstract regions can come from several different processes, we must specify how the different attributes of the different processes will be combined. We have tried two different forms of combination: 1) treat the different types of regions independently and combine only at the time of classification ($p(o|\{F_I^a\}) = \prod_a p(o|F_I^a)$) and 2) form intersections of the different types of regions and use them, instead of the original regions, for classification. In the first case, only the specific attributes of a particular type of region are used for the respective mixture models. If a set of regions came from a color segmentation, only their color attributes (HSI) are used, whereas if they came from a texture segmentation, only their texture coefficients are used. In the second case, the intersections are smaller regions with properties from all the different processes. Thus an intersection region would have both color attributes and texture attributes.

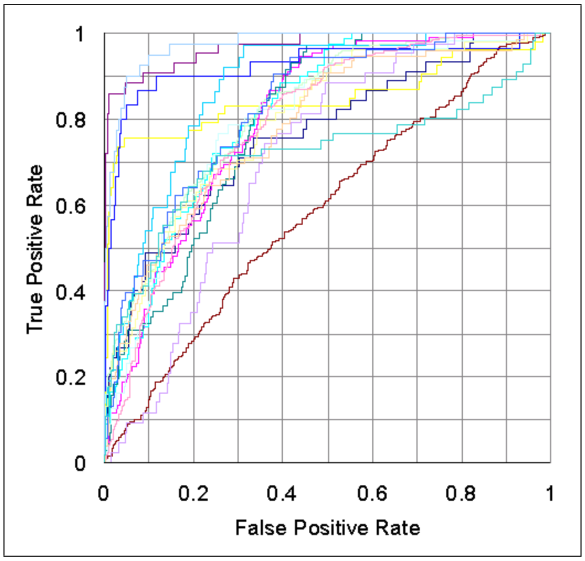

Figure 2. ROC curves for the 18 object classes with independent treatment of color and texture.

Our test database of 860 images was obtained from two image databases: creatas.com and our groundtruth database http://www.anonymous. The images are described by 18 keywords. The keywords and their appearance counts are: mountains (30), orangutan (37), track (40), tree trunk (43), football field (43), beach (45), prairie grass (53), cherry tree (53), snow (54), zebra (56), polar bear (56), lion (71), water (76), chimpanzee (79), cheetah (112), sky (259), grass (272), tree (361). We ran a set of cross-validation experiments in each of which 80% of the images were used as the training set and the other 20% as the test set. Figure 2 illustrates the ROC curves (true positive rate vs. false positive rate) for each object, treating color and texture independently. Figure 3 illustrates the results for the same objects, using intersections of color and texture regions. In general, the intersection method achieves better results than the independent treatment method. This makes sense because, for example, a single region exhibiting grass color and grass texture is more likely to be grass than one region with grass color and another with grass texture. Using intersections, most of the curves show a true positive rate above 80% for false positive rate 30%. The poorest results are on object classes “tree,” “grass,” and “water,” each of which has a high variance, for which a single Gaussian model is not sufficient.

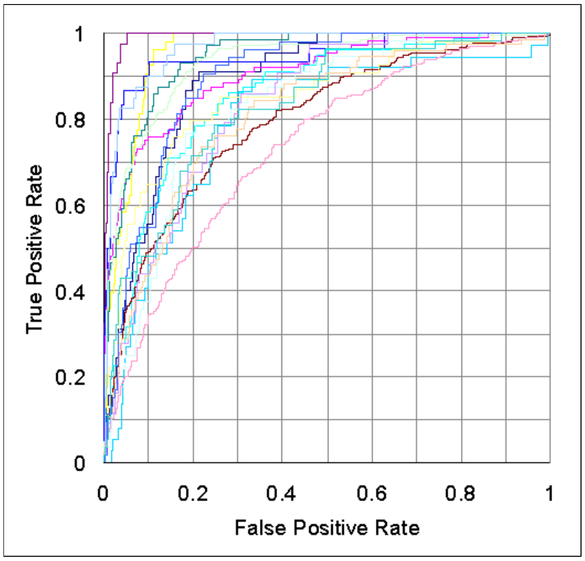

Figure 3. ROC curves for the 18 object classes using intersections of color and texture regions.

Figure 4 shows the top three images returned for several different object classes. The football image is an example of a false positive for the cherry tree class; the crowd has roughly the same color and texture as a cherry tree.

4. Conclusions

We have presented a new method for recognizing classes of objects in color photographic images of outdoor scenes. We represent images as sets of abstract regions and model each of these regions as a mixture of Gaussians. We developed a new semi-supervised EM-like algorithm that is given the set of objects present in each training image, but does not know which regions correspond to which objects. Our EM variant is able to break the symmetry in the initial solution. We have tested the algorithm on a dataset of 860 hand-labeled color images. We compared two different methods of combining different types of abstract regions, one that keeps them independent and one that intersects them. The intersection method had a higher performance as shown by the ROC curves in our paper. While these preliminary results are promising, they are not yet good enough. Since regions of high variance are not well modeled by a single Gaussian distribution, we plan to try multiple Gaussians for each object class and have had some initial success with trees. Since color and texture are not powerful enough for many objects, we plan to add structure regions that can help recognize buildings and other man-made structures. We also plan to use the spatial relationships among regions, which is another important factor in recognition. Finally, due to the large number of possible object classes, we anticipate the eventual need for a hierarchy of classifiers to handle the load.

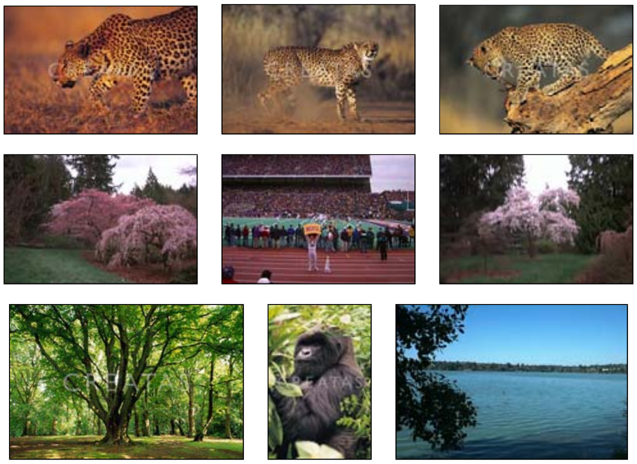

Figure 4. The top 3 test results for cheetah, cherry tree, and tree.

References

[1] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Region-based image querying. *Proceedings of the 1997 IEEE Workshop on Content-Based Accesss of Image and Video Libraryies*, pages 42–49, June 1997.

[2] R. Fergus, P. Perona, and A. Zisserman. Object-class recognition by unsupervised scale-invariant learning. *CVPR*, 2:264–271, 2003.

[3] D. Forsyth and M. Fleck. Body plans. *CVPR*, pages 678–683, 1997.

[4] D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *IJCV*, 5(2):195–212, 1990.

[5] Y. Li and L. G. Shapiro. Consistent line clusters for building recognition in cbir. *ICPR*, pages 952–6, 2002.

[6] W. Y. Ma and B. S. Manjunath. Netra: A toolbox for navigating large image databases. *ICIP*, 1997.

[7] H. Murase and S. K. Nayar. Visual learning of object models from appearance. *IJCV*, 1992.

[8] J. Shi and J. Malik. Normalized cuts and image segmentation. *CVPR*, pages 731–737, 1997.

[9] J. R. Smith and C. S. Li. Image classification and querying using composite region templates. *CVIU: Special Issue on Content-Based Access of Image and Video Libraries*, 75(1-2):165–174, 1999.

[10] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 1991.

[11] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. *ECCV*, pages 18–32, 2000.