Image Background Search: Combining Object Detection Techniques with Content-Based Image Retrieval(CBIR) Systems

Rohini Srihari, Zhongfei Zhang, Aibing Rao
CEDAR, SUNY At Buffalo
rohini, zhongfei, arao@cedar.buffalo.edu

Abstract

A framework for combining object detection techniques with a content-based image retrieval(CBIR) system is discussed. As an example, a special CBIR system which focuses on human faces as foreground and decides the similarity of images based on background features is presented. This system may be useful in automatically generating albums from consumer photos.

Keywords: Content-Based Image Retrieval(CBIR), Averaged Precision and Recall(APR), Performance Area, Background Retrieval, Foreground Retrieval.

1. Introduction

Traditional CBIR systems [2, 5, 1, 4] have achieved great success when dealing with pure scenery image data, but when images change from simple scenery to more semantically complex domain, such as images with human beings, the performance reduces significantly. The reason is: (i). These systems do not differentiate the semantic foreground objects (e.g. human beings) from background scenery; (ii). The matching schemes are based on certain primitive statistical information extracted from the combination of the foreground and the background scene. Examples of these systems may be found in [2, 6, 5, 1, 4]. To improve the performance of a CBIR system on semantically complex images, such as consumer photos, a more intelligent approach to picture classification and retrieval is necessary.

In this paper, we first review the results of a traditional CBIR system which uses histogram vectors as features, and then we compare the results for two types of database: a simple database containing only senery images and a typical consumer photo database containing images with human faces. Experiments show that the system works well for the first database but fails to perform reasonably well for the second database.

Based on this observation, we propose an image retrieval framework that partitions an image into foreground and background. This phenomenon is frequently true for consumer photos and news pictures, in which there are semantic objects serving as foreground superimposed against a background scene. Furthermore, for most of the foreground objects, we can further classify them into a limited number of semantic types (e.g. human beings, airplanes, buildings, etc.), called *prototypes*. Thus, assuming that we have reliable tools to detect these objects, we can focus the retrieval problem on either the background or foreground scene. Since human beings are considered as one of the most frequently encountered semantic objects in consumer phtos as well as news photos, as an example, we propose a testing system that focuses on human beings as the foreground scene and conducts the similary retrieval based on the background scene.

2. Performance Comparison of a Traditional CBIR system on Different Test Image Databases

To give an idea about the performance difference of a traditional CBIR system on various types of databases, we have done extensive experiments, the results are illustrated below.

Database Preparation. The images are contributed by Eastman Kodak as part of the consumer photo samples. Two classes of images are selected: *Pure Scenery Images* and *Images with Human Faces*. In the following, we use *PSI* and *HFI* to denote these two classes respectively. There are *58* images for *PSI* and *66* images for *HFI*. In both classes, images are divided into four semantic groups: (1). *Watery Scene:* Images about river, boat, etc.; (2). *Street Scene:* Images about street, house, etc.; (3). *Indoor Scene:* Images about furniture, baby, etc.; (4). *Mountain and Sky:* Images of mountains under the sky, with or without humans. Combining these images, we set up three databases: *PSI*(58 images), *HFI*(66 images) and *TOTAL*(*PSI+HFI*, 124 images).

Performance Evaluation. The CBIR system first generates a color histogram vector for each image. Then for each

data set, with each image as a query, an ranking is generated and meanwhile the *precisions* and *recalls* based on above similarity groups are calculated for top 10 matches and then the *averaged precision and recall(APR)* is calculated. The results are shown in *Figure 1*.

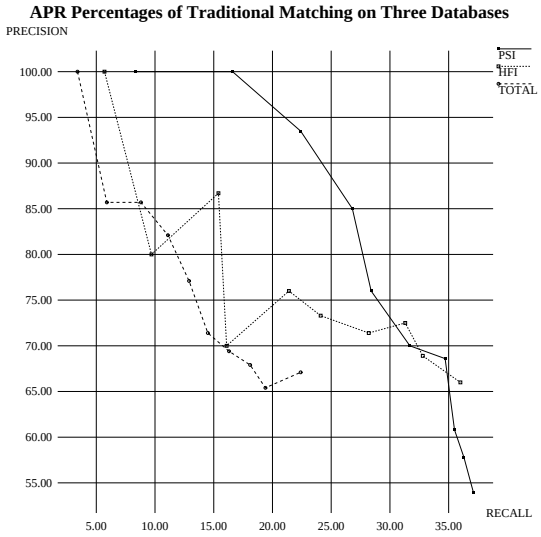

Figure 1. Averaged Precision-Recall curves for different databases. The system performs the best when the test data set is pure scenery. Number of retrieved images: 10.

Numerically, we will use the *area* enclosed by a APR curve and the axises as a performance metric, called *performance area*, it is defined as

$$\frac{1}{2}\sum_{i=1}^{N-1}(x_{i+1}-x_i)(y_{i+1}+y_i)$$

Where (x_i, y_i) is the (recall, precision) pair when the number of retreived images is i and N is the total number of top matches. In *Figure 1*, the areas for the curves are: $2915.82(PSI)$, $2292.97(HFI)$ and $1471.39(TOTAL)$. So when database changes from pure scenery(PSI) to human beings(HFI), the performance area reduces nearly 21.4%, if it further changes to a mixed one($TOTAL$), the performance area reduces 49.5%! This is the motivation to propose the framework of a theory of image similarity retrieval based on foreground and background partition, which is introduced in the next section.

3. A Framework: Combining Object Detection with CBIR Systems

Traditional CBIR systems represent image contents at a primitive statistics level. If a statistical feature is extracted globally, the relationship between the extracted primitive features and the contents of an image is very weak, especially for images containing semantic foreground objects. The performance can be improved by local matching through some brute force partitions such as quadtree, tessellation and so on. However, it still suffers from the weak correlation between the statistical features and the semantics of an image.

Object detection attempts to establish the connection between the raw image data and its semantic contents. By combining object detection techniques with an existing CBIR system, image retrieval can be expected to move from the lower level statistical approach to a higher level semantic approach. Herin, we present a representation of image features which combines the semantic features(e.g. objects) and the primitvie statistical features(e.g. histograms).

Let $FS = \{< F_i, S_i >\}_{i=1}^{N}$ be a set of $< feature\ function,\ similarity\ measure >$ pair. Typical F_i can be color histogram, texture feature, shape descriptor, etc., and S_i is the similarity measure developed for feature F_i . Let $OD = \{od_j\}_{j=1}^{M}$ be a set of object detectors such as face detector, building detector, etc.

For a given image I, apply each of the object detectors on I, resulting in a sequence of objects. For each object detected, a vector is formed, called an *object subimage* defined as an instance of type

Record:
int: col, row; *dimension of the subimage*
int: X0, Y0; *Left-top corner coordinates*
Model: model; *priori model of the object*

where **Model** is a prototype of the object based on the primitive features. Then the image I is partitioned into a vector of subimage records,

$$(R_1, R_2, ..., R_K, B)$$

where B is the *background* subimage which is the remaining portion of I after cutting off all the object subimages and

$$R_j = (O_j^1, O_j^2, ..., O_j^{n_j})$$

is a list of object subimages detected by the j^{th} detector while n_j is the number of objects of this type. Note that n_j may be zero for some j indicating that od_j detected no objects. Next, we discuss similarity measures using this representation.

Given two images p, q, following above process, we have two feature vectors, $(R_1^p, R_2^p, ..., R_K^p, B^p)$ and $(R_1^q, R_2^q, ..., R_K^q, B^q)$ with

$$R_j^p = (P_j^1, P_j^2, ..., P_j^{n_j^p});\ \ R_j^q = (Q_j^1, Q_j^2, ..., Q_j^{n_j^q})$$

Then there are two choices to define similarity of p and q:

(1). *Background Similarity* By ignoring all object subimages, the similarity between p and q is defined as

$$S_B(p,q) = \mathcal{S}(S_1(F_1(B^p), F_1(B^q)), ..., S_N(F_N(B^p), F_N(B^q)))$$

where S_i is the similarity measure developed for F_i and $\mathcal{S}$ is an accumulative function which accumulates the similarity on each feature dimension into a final measure. Image retrieval based on this similarity is called *background retrieval*.

(2). *Foreground Similarity* To explore similarity measure incorporating object subimages, we first consider the distance of image p and q for each object prototype, namely, the distance, $m_j(R_j^p, R_j^q)$, between R_j^p and R_j^q for each $j = 1, 2, ..., K$. Possible choices are:

- *Object Count Difference:* difference of the number of objects of the same model, i. e.

$$m_j(R_j^p, R_j^q) = |n_j^p - n_j^q|$$

- *Object Spatial Configuration Difference:* difference of the spatial relationships of the objects

$$m_j(R_j^p, R_j^q) = \mathcal{M}_j(\mathcal{G}(R_j^p), \mathcal{G}(R_j^q))$$

  where $\mathcal{G}$ is a function of the list of all object subimages of a same prototype, called *spatial configurations* of these objects, representing relative positions, ratio of areas or whatever reasonable quantities for comparing objects. Due to the representation of an object subimage, it is in fact a function of the coordinates, the dimensions, as well as the prototype instances of all object subimages of the same prototype. $\mathcal{M}_j$ is a distance measure between the spatial quantities calculated by $\mathcal{G}$.

Then a *foreground similarity measure* can be defined as,

$$S_F(p, q) = \mathcal{C}(m_1(R_1^p, R_1^q), ..., m_K(R_K^p, R_K^q))$$

where $\mathcal{C}$ is an accumulative function, which may be linear or non-linear. When $\mathcal{C}$ is linear, it assigns weights to different objects. When $\mathcal{C}$ is non-linear, it may be very complicated. For example, it may be designed to take some object detectors as filters and others as retrieval operators.

Moreover, by combining background and foregound similarities, a general form of similarity between p and q is

$$S(p, q) = \mathcal{A}(S_F(p, q), S_B(p, q))$$

where $\mathcal{A}$ is a linear or non-linear accumulative function. If it is linear, it assigns weights to both similarities. when the weight of the foreground similarity is zero, it reduces to background retrieval, and similarly for the foreground retrieval. The designs of all these accumulative functions are application-specific; in applications of consumer photo classification, queries may be either in favor of the foregound matching, or in favor of the backgound matching.

4. An Example: Background Matching by Combining Face Detection with a CBIR System

As an example of the framework proposed above, we have implemented an image retrieval system based on background matching by combining face detection with a traditional CBIR system discussed in Section 2. We only need to modify the feature generation process. The modification follows these steps to generate feature vectors for each image in the database:

Step 1: Run face detection algorithm to detect faces;

Step 2: Generate the background subimage by cropping off the faces together with the bodies of the human beings. In more detail, suppose $\{(C, R), (X, Y), M\}$ is a face record generated by the face detection, where (C, R) are the dimensions of the face subimage, (X, Y) are the coordinates of the left-top corner, and M is a face model. Notice that a human body should appear along with the face and the width of a human body is normally *twice* (more accurate quantity may be obtained by statistics) as the width C of the face, we obtain a body subimage: $\{(max(0, C - \frac{X}{2}), R + Y), (2 * X, H - R - Y)\}$, where H is the height of the original image, the first pair is the coordinate of the left-top corner and the second pair is the dimension. The body subimage is the box underneath the face with a width as twice as the face and with a height from the bottom edge of the face subimage to the bottem edge of the orginal image. A special case arises when a body box is outside of the original image. After cutting off all face subimages as well as associated body boxes, the remaining portion is a background subimage.

Step 3: Generate features for the subimages and use them as feature vectors for retrieving the global images. The top matches should have similar background as the query image.

Figure 2. An input image to the background extraction algorithm.

Figure 2 is an example, where the face detection algorithm detects two face records: $\{(218, 76), (28, 24), \mathcal{F}\}$ and $\{(292, 44), (40, 40), \mathcal{F}\}$, implying that two objects (i.e. faces) of model $\mathcal{F}$ (i.e. face model) are detected; one is a 28×24 subimage starting at $(218, 76)$, and the other is a 40×40 subimage starting at $(292, 44)$. After cropping off these faces and their associated body boxes, the background image is shown in *Figure 3*. The primitives feature of this subimage are regarded as the features of the original image for indexing and retrieval.

Figure 3. The background image of the image in Figure 2. Faces and the heuristic body boxes are cropped out.

Performance Evaluation. We continue to use the images and the database notations in Section 2. For each image in database *HFI*, we apply a face detection system [8, 7] to detect faces. For those images where the face detector failed, we manually correct the face detection. In this work, our focus is on giving an example to show the framework of this image retrieval theory, rather than showing how robust our face detection system works. In practice, we can employ any of the existing face detection system in the literature (e.g. [3]).

After applying this retrieval method to database *HFI*, we can compare the effectiveness of this method with that of the traditional one. Their APRs are shown in *Figure 4*. Notice that the performance area is improved 10% on HFI. For database *TOTAL* which combines pure scenery data and images with human faces, we can also use this background matching method to improve the performance. The APRs for this new method and the traditional method for the database *TOTAL* are shown in *Figure 5*. The performance area is improved 20% on $TOTAL$.

From *Figure 4* and *Figure 5*, we can see that the new background matching technique greatly improves the performance on both databases.

As an example, we take an image in the database *Total* as a query, then use the traditional matching and the back-

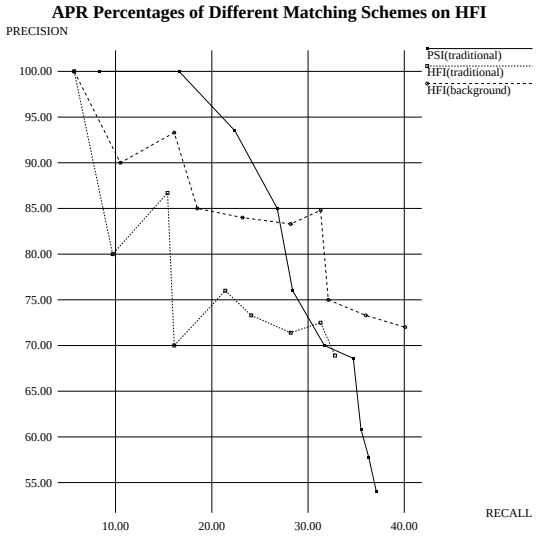

Figure 4. Average Precision-Recall for two different schemes on HFI. The background matching greatly improves the performance on HFI; it approximates and then outperforms the performance of traditional matching on PSI. Number of retrieved images: 10. The result of the original system on PSI is also shown for comparison purpose.

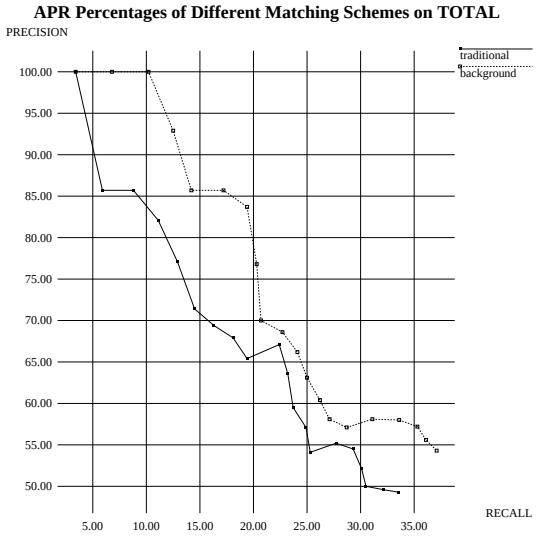

Figure 5. Average Precision-Recall for different matching schemes on database TOTAL. The background matching scheme greatly improves the performance on database TOTAL. Number of retrieved images: 20.

Figure 6. Top 6 matches by using the top-left image to query database TOTAL with traditional approach. The 4^{th}, 5^{th} and 6^{th} matches are false-positive because the boy happens to have a similar color to the sky.

ground matching to query the same database TOTAL, the top 6 matches are shown in *Figure 6* and in *Figure 7*, respectively. The explanation for the great difference of the

Figure 7. Top 6 matches by using the top-left image to query TOTAL with background matching. All 6 matches are real similar ones. The performance is greatly improved, compared to the result in Figure 6.

performances of this example is that in the query image, the region color of the boy happens to be very similar to the color of the sky and this causes that some images with sky are also retrieved in highest ranks in traditional matching. In background matching, the boy's region is ignored so that false positives are reduced.

5. Conclusion

By combining more intelligent tools such as object detection techniques with an existing content-based image retrieval system, the performance can be improved significantly for some specific tasks such as image classifications according to the background. The idea makes the image retrieval techniques move from purely statistical domain to semantic domain. The performance of such a new system, on the other hand, is based on the assumption that there is a reliable object detection system available for certain type of objects. It is clear that with the current progress of the research in computer vision and image understanding, such robust object detection techniques are available in certain application areas (such as face detection), which makes possible for the successful application of this new technique in image retrieval.

References

[1] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. Shu. The virage image search engine: An open framework for image management. *Symposium on Electronic Imaging: Science and Technology — Storage & Retrieval for Image and Video Databases IV*, 2670:76–87, 1996.

[2] R. Barber, M. Flickner, J. Hafner, W. Niblack, and D. Petkovic. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3:231–262, 1994.

[3] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *Trans. on PAMI*, 20(1), 1998.

[4] S. Servetto, Y. Rui, K. Ramchandran, and T. S. Huang. A region-based representation of images in mars. *Special issue on multimedia Signal Processing, Journal on VLSI Signal Processing Systems*, October 1998.

[5] J. Smith and S.-F. Chang. Visualseek: A fully automated content-based image query system. *Proceeding ACM International Conference of Multimedia*, pages 87–98, November 1996.

[6] J. R. Smith. *Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression.* PhD thesis, Columbia Univ., 1997.

[7] R. K. Srihari and Z. Zhang. Finding pictures in context. *Proc. of IAPR International Workshop on Multimedia Information Analysis & Retrieval, Springer-Verlag Press*, pages 109 – 123, 1998.

[8] Z. Zhang. **Invited paper:** recognizing human faces in complex context. In *Proc. of the International Conference on Imaging Science, Systems, and Technology*, pages 218 – 225. CSREA Press, 1998.