CATEGORY-BASED IMAGE RETRIEVAL

S. Newsam, B. Sumengen, and B.S. Manjunath
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106-9560
snewsam,sumengen,manj@ece.ucsb.edu

ABSTRACT

This work presents a novel approach to content-based image retrieval in categorical multimedia databases. The images are indexed using a combination of text and content descriptors. The categories are viewed as semantic clusters of images and are used to confine the search space. Keywords are used to identify candidate categories. Content-based retrieval is performed in these categories using multiple image features. Relevance feedback is used to learn the user's intent—query specification and feature-weighting—with minimal user-interface abstraction.

The method is applied to a large number of images collected from a popular categorical structure on the World Wide Web. Results show that efficient and accurate performance is achievable by exploiting the semantic classification represented by the categories. The relevance feedback loop allows the content descriptor weightings to be determined without exposing the calculations to the user.

1. INTRODUCTION

Indexing diverse collections of multimedia data remains a challenging problem. Even though significant progress has been made toward developing effective content-descriptors, evidenced by the forthcoming MPEG-7 standard, it is still difficult to bridge the gap between low-level image analysis and image understanding at the semantic level. This gap limits access solutions since users usually interact at the semantic level.

The images found on the World Wide Web (WWW) are a prime example of a multimedia collection that is difficult to index. Low-level features, such as color and texture, can be extracted and used for similarity searches. The results might be *visually* relevant but it is unreasonable to expect them to be *conceptually* relevant. For this, the content-based searches must be constrained to semantically relevant sets of images. Due to the size of the dataset, manual classification is not feasible.

This work investigates how existing semantic structure can be exploited by a multimedia access system even if this structure is not perfect. Our approach recognizes that the WWW is not just a large collection of images with loosely associated text but there is existent structure which can be exploited. The work also demonstrates how relevance feedback can refine query intent using a simple and intuitive interface.

The objective is to index a large number of images from the WWW for efficient and accurate access given the following constraints:

- Image discovery and indexing should be automatic.
- Image search should be semantically constrained.
- The search interface should be simple and intuitive.

The proposed Categorical Image Search (CIS) system accomplishes this by:

- Using an existing directory for image discovery.
- Using the directory to constrain content-based searches.
- Using relevance feedback to learn query intent.

The rest of the paper is organized as follows. Sections 2 and 3 describe image ingestion and image search, respectively. Section 4 presents some preliminary results and discussions.

2. IMAGE INGESTION

Images are collected by spidering websites listed in the DMOZ Open Directory Project [1]. The DMOZ project's goal is to "produce the most comprehensive directory of the web, by relying on a vast army of volunteer editors." A total of 2,633,071 sites in 369,307 categories are managed by 36,786 editors. A custom web-spider locates images at sites parsed from a snapshot of the directory. Irrelevant images, such as icons and banners, are filtered-out. Different types of text associated with the images are stored in a relational database. This includes category names, image names, image ALT tags and HTML text. Thumbnail versions of the images are created for display purposes. Finally, texture and color content descriptors are extracted.

Texture descriptors are extracted by applying a set of Gabor-wavelet filters tuned to combinations of three scales and four orientations [2]. The feature vector components are the means and standard-deviations of the outputs of the 12 filters. The results in a 24-dimension texture feature vector. The similarity between images in the texture feature space is computed using the L1 norm.

Color descriptors are derived from the color distribution histogram computed in the CIE $L^*u^*v^*$ color space. The three color dimensions are quantized to four levels resulting in a 64-dimension color feature vector. The similarity between images in the color feature space is computed using the Euclidean distance.

3. DATABASE ORGANIZATION AND IMAGE SEARCH

The text in the relational database is compiled to create an extensive keyword list. This list is the basis of an inverted-index of the categories based on keyword frequency. The other major component of the database is the collection of image features. Each image is represented by the 88 numbers from the texture and color feature vectors. The image features are organized by category. Figure 1 shows the structure of the database. Each keyword can index any number of categories but only the top 10 are retrieved during a lookup. A category can be indexed by multiple keywords.

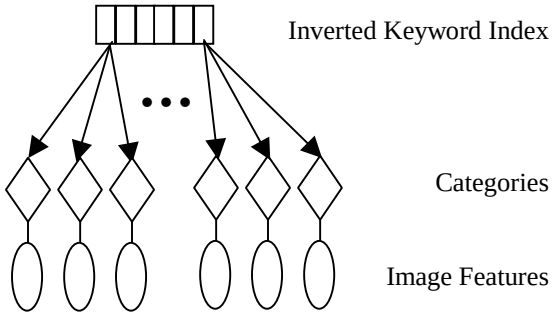

Figure 1 – Database Structure

Image searches consists of two steps. First, a text-based search identifies the categories. Then, a constrained content-based search with relevance feedback is performed among the images from these categories.

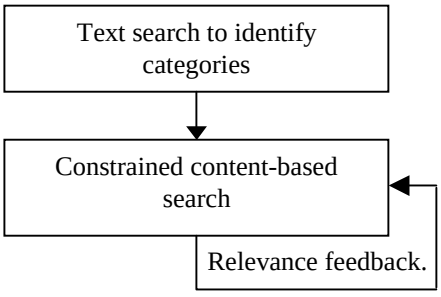

Figure 2 – Image Search

3.1. Text Search

Image searches are initiated with keywords. The keywords determine which categories to confine the content-based search to. The inverted index is used to retrieve the image feature clusters corresponding to these categories. Constraining the content-based search improves the semantic relevance of the results and allows the search to be performed in real time

3.2. Constrained Content-based Search

The content-based search uses a query-by-example paradigm. The user first selects one or more images from a set chosen randomly from the candidate categories. This selection is used to perform a similarity search in the combined feature spaces. If the user is not satisfied with the results, he/she can add or remove images and perform additional similarity searches. A relevance feedback loop is used to iteratively refine the query vector and search space.

3.3. Relevance Feedback

Relevance feedback is used to compute the query vector, compute the relative feature weightings, and prune the search space. This part of the search only uses the image descriptors. The keywords from the first stage of the search are no longer used and the categories only serve to constrain the search.

Query Specification: If the user selects a single image then its 64-dimension color and 24-dimension texture feature vectors are concatenated to produce the 88-dimension query vector used in the content-based search. If multiple images are chosen then their feature vectors must be combined to produce a single 88-dimension query vector. If it is assumed that the user selects a visually similar set of images then a straightforward solution is to use the average of the vectors. In the CIS system, the 88-dimension query vector is computed to be the average of the individual vectors in the case where multiple images are chosen.

Relative Feature Weightings: The distribution of the user-selected images is used to determine the relative feature weightings; i.e., which feature, color or texture, is more important to the user for a particular search. Intuitively, the feature space in which the images are more "tightly clustered" should be weighed more. For a set R of example images:

$$\bar{d}_{texture} = \frac{1}{|R|} \sum_{i,\, j \in R} d_{texture}(i, j)$$

and

$$\bar{d}_{color} = \frac{1}{|R|} \sum_{i,\, j \in R} d_{color}(i, j).$$

The feature weightings are then:

$$w_{texture} = \frac{1}{\bar{d}_{texture} + \varepsilon} \text{ and } w_{color} = \frac{1}{\bar{d}_{color} + \varepsilon}$$

where ε is a small value to prevent one of the features from becoming too significant. The final distance measure used to rank-order the N nearest neighbors is:

$$d(\cdot,\cdot) = w_{texture} d_{texture}(\cdot,\cdot) + w_{color} d_{color}(\cdot,\cdot).$$

This application of relevance feedback is similar to that presented in [4].

If the user selects only one image then there is not enough knowledge to determine which feature, color or texture, is more significant. In this case, two similarity searches are performed: an N/2 nearest neighbor search using texture descriptors and an N/2 nearest neighbor using color descriptors.

Category Pruning: After several iterations of the relevance feedback loop, the search space is further reduced to just those categories from which the user is selecting images. This improves the relevance of the results.

4. PRELIMARY RESULTS AND DISCUSSIONS

An implementation of the CIS system indexes over 600,000 images from 38,604 sites in 1,623 categories. It is available online in the Demos section at http://vision.ece.ucsb.edu.

Figures 3 through 5 show the different steps of a search initiated using the keyword "hat." In this case, the inverted index is used to constrain the content-based search to approximately 9,000 images in 10 categories such as Shopping/Clothing/Hats, Shopping/Clothing/Sportswear, and Shopping/Clothing/Costumes. From this point on, the search only uses the image-derived feature descriptors. Figure 3 shows 12 images displayed at random from the constrained set. The user selects the top right image and initiates a *constrained* content-based search. As discussed above, the relative feature weightings cannot be determined if only a single image is selected so two nearest neighbor searches are performed. The top row of Figure 4 shows the query image and the 5 most similar images with respect to the color descriptor. The bottom row shows the query image and the 5 most similar images with respect to the texture descriptor. The user adds the bottom right image and initiates another content-based search. The two selected images are used to determine the query vector and the relative feature weightings, as discussed above. The final results are shown in Figure 5. If necessary, the user can continue to add or remove images and perform additional content-based searches.

This example shows that the CIS system retrieves *visually* and *conceptually* relevant images within a few iterations. The *integration* of the semantic classification provided by the categories and the visual similarity retrieval capabilities of the image descriptors makes this possible. Neither one of these techniques alone provides relevant results. Figure 3 shows that fewer than half the images chosen at random from the constrained set are related to the concept "hat." This is because the semantic classification provided by the categories is not perfect. Figure 6 shows the results of an *unconstrained* content-based search using the same query vector and feature weightings as the search in Figure 5. As expected, the concept of a "hat" is lost when the content-based search is over the entire database of 600,000 images.

Even though the nearest neighbor searches are only performed on a subset of the database, they are still computationally expensive due to the high-dimensionality of the image feature space. Conventional indexing methods, such as the R-tree and its variants, cannot be used because the similarity metric is not fixed. A novel algorithm has been developed to address this problem [5]. The method exploits the correlations between consecutive nearest neighbor searches to filter out candidate matches. This greatly reduces the number of I/O accesses in the database. The method has been shown to be effective for the 600,000 images indexed by the CIS system.

Future work includes a quantitative evaluation of the CIS system. Determining query recall is difficult since establishing ground truths for the entire dataset is impractical. One approach being investigated is analyzing the validity of the feature clusters that are derived from the category structure.

5. ACKNOWLEDGEMENTS

This research is supported in part by an NSF grant #IRI-9704785, an ONR/AASERT #N00014-98-1-0515, and by Samsung Electronics.

6. REFERENCES

[1] The DMOZ Open Directory Project. http://dmoz.org.

[2] Manjunath, B.S. and Ma, W.Y. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, (no.8), IEEE Comput. Soc., Aug. 1996. p.837-42.

[3] Manjunath, B.S., Wu, P., Newsam, S., and Shin, H.D. A Texture Descriptor for Browsing and Similarity Retrieval. *Journal of Signal Processing: Image Communication*, Volume 16, Issue 1-2, page 33-43, September 2000.

[4] Rui, Y., Huang, T.S., Ortega, M., and Mehrotra, S. Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8,(no. 5), IEEE Comput. Soc., September 1998. p.644-55.

[5] Wu, P. Ph.D. Dissertation, University of California at Santa Barbara, in preparation.

Figure 3 –Random images from categories identified using the keyword “hat.” The top right image is selected for a constrained content-based search.

Figure 4 – Results of a *constrained* content-based search using the image selected in Figure 3. Top row shows query image and 5 most similar images with respect to the color descriptor. Bottom row shows query image and 5 most similar images with respect to the texture descriptor. The bottom right image is added to the query.

Figure 5 – Results of *constrained* content-based search using the two images selected in Figure 4. Relevance feedback is used to compute the query vector and relative feature weightings.

Figure 6 – Results of an *unconstrained* content-based search using the same query vector and feature weightings as the search in Figure 5.