CONTENT-BASED IMAGE RETRIEVAL WITH RELEVANCE FEEDBACK IN MARS

Yong Rui, Thomas S. Huang and Sharad Mehrotra

Beckman Institute and Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
E-mail:yrui@ifp.uiuc.edu, huang@ifp.uiuc.edu and sharad@cs.uiuc.edu

ABSTRACT

Technology advances in the areas of Image processing *(IP) and* Information Retrieval *(IR) have evolved separately for a long time. However, successful content-based image retrieval systems require the integration of the two. There is an urgent need to develop integration mechanisms to link the image retrieval model to text retrieval model, such that the well established text retrieval techniques can be utilized.*

Approaches of converting image feature vectors (IP domain) to weighted-term vectors (IR domain) are proposed in this paper. Furthermore, the relevance feedback technique from the IR domain is used in content-based image retrieval to demonstrate the effectiveness of this conversion. Experimental results show that the image retrieval precision increases considerably by using the proposed integration approach.

1. INTRODUCTION

Technology advances as well as the emergence of large scale multimedia applications have made development of effective techniques for visual and multimedia retrieval system one of the most challenging and important directions of the future research. Such systems will support visual data retrieval based on their rich internal contents. Development of such retrieval systems requires the integration of various techniques in the fields of *Image Processing* (IP) and *Information Retrieval* (IR).

Traditionally, the above two research areas have been studied in isolation with little or no interaction. Although effective algorithms for image feature extraction and representation have been developed in IP, how these algorithms can be incorporated in the visual data management system to support effective retrieval has not been fully explored. On the other hand, although the retrieval system framework and various retrieval techniques have been established in IR, research has primarily focussed on textual data and has not considered image or other visual media.

The isolation of these two research areas is further evidenced by each area's research literature. Most of the existing image retrieval systems [1, 2] have not made use of the retrieval techniques developed in IR. On the other hand, most existing IR systems [3, 4] focus solely on the text-based information.

We believe that effective content-based retrieval of image and multimedia data requires an integration of the retrieval models developed in the IR literature and the feature extraction and representation methods developed in IP area. To illustrate the usefulness of such an integration, in this paper we adopt the term weighting and relevance feedback techniques to content-based image retrieval. This integration approach has been implemented in MARS [1]. In section 2, term weighting and relevance feedback techniques will be briefly reviewed. Image texture feature representations are briefly described in section 3. In section 4, how to convert the image feature vectors to IR weight vectors is addressed in detail. Experimental results and conclusions will be given in sections 4 and 5 respectively.

2. TERM WEIGHTING AND RELEVANCE FEEDBACK

An IR model consists of a document model, a query model, and a model for computing similarity between the documents and the queries. One of the most popular IR models is the vector model[14, 3, 15]. Various effective retrieval techniques have been developed for this model. Among them, *term weighting* and *relevance feedback* are of fundamental importance.

2.1. Term Weighting

Term weighting is a technique of assigning different weights for different keywords (terms) according to their relative importance to the document [15, 3].

If we define w_{ik} to be the weight for term $t_k, k = 1, ..., N$, in document i, where N is the number of terms, document i can be represented as a weight vector in the term space:

$$D_i = [w_{i1}; ...; w_{ik}; ...; w_{iN}] \quad (1)$$

To correctly estimate the weights, we need to consider two aspects. First, if term k is frequently occurred in the

This work was supported in part by NSF/DARPA/NASA DLI Program under Cooperative Agreement 94-11318, in part by ARL Cooperative Agreement No. DAAL01-96-2-0003, and in part by CSE Fellowship, UIUC.

[1]Multimedia Analysis and Retrieval System (MARS) [5, 6, 7, 8, 9, 10, 11, 12, 13] is the content-based multimedia retrieval system developed at the University of Illinois at Urbana-Champaign. URL: http://jadzia.ifp.uiuc.edu:8000.

0-8186-8183-7/97 $10.00 © 1997 IEEE

document i, then w_{ik} should be assigned high value. This intuition suggests that a term frequency (*tf*) factor should be included in the estimation of w_{ik}. Second, *tf* alone cannot ensure an acceptable estimation. When the high frequency term is not concentrated in a few documents, but instead spreading over all documents, we should give this term low weight. This introduces the well-known inverse document frequency (*idf*), which varies inversely with the number of documents in which a term appears.

$$idf_k = log_2 \frac{M}{df_k} + 1 \tag{2}$$

where df_k is the document frequency for term k and M is the total number of documents in the collection. Experiments have shown that the product of *tf* and *idf* is a good estimation of the weights[14, 3, 15].

The query Q has the same model as that of document D, i.e. it is a weight vector in the term space:

$$Q = [w_{q1}; ...; w_{qk}; ...; w_{qN}] \tag{3}$$

The similarity between D and Q is defined as the Cosine distance.

$$Sim(D, Q) = \frac{D\ Q}{||D||\ ||Q||} \tag{4}$$

where $||\ ||$ denotes norm-2.

2.2. Relevance Feedback

As we can see from the previous subsection, in the vector model, the specification of w_{qk}'s in Q is very critical, since the similarity values ($Sim(D, Q)$'s) are computed based on them. However, it is usually difficult for a user to map his information need into a set of terms precisely. To overcome this difficulty, the technique of *relevance feedback* has been proposed [3, 15, 14]. Relevance feedback is the process of automatically adjusting an existing query using information fed-back by the user about the relevance of previously retrieved documents.

The mechanism of this method can be described elegantly in the vector space. If the sets of relevant documents (D_R) and non-relevant documents (D_N) are known, the optimal query can be proven to be[14, 3, 15]

$$Q_{opt} = \frac{1}{N_R} \sum_{i \in D_R} D_i - \frac{1}{N_T - N_R} \sum_{i \in D_N} D_i \tag{5}$$

where N_R is the number of documents in D_R and N_T the number of the total documents.

In practice, D_R and D_N are not known in advance. However, the relevance feedback obtained from the user furnishes approximations to D_R and D_N, which are referred as, D'_R and D'_N.

The original query Q can be modified by putting more weights on the relevant terms and less weights on the non-relevant terms.

$$Q' = \alpha Q + \beta(\frac{1}{N_{R'}} \sum_{i \in D'_R} D_i) - \gamma(\frac{1}{N_{N'}} \sum_{i \in D'_N} D_i) \tag{6}$$

where α, β and γ are suitable constants[3, 15]; $N_{R'}$ and $N_{N'}$ are the numbers of documents in D'_R and D'_N. Q' approaches Q_{opt}, as the relevance feedback iteration moves on. Experiments show that the retrieval performance can be improved considerably by using relevance feedback [14, 3, 15].

3. IMAGE TEXTURE FEATURE REPRESENTATIONS

We will use texture feature based image retrieval in this paper to show how to integrate the techniques in IP and IR. To demonstrate the validity of the proposed approach, we will convert two well known texture representations to the weighted-term vector model. We briefly describe the two representations in this section.

3.1. Wavelet representation

An input image is fed into a wavelet filter bank and is decomposed into de-correlated sub-bands. Each sub-band captures the feature of some scale and orientation of the original image.

Specifically, we decompose an image into three wavelet levels; thus having 10 sub-bands. For each sub-band, the standard deviation of the wavelet coefficients is extracted. The 10 standard deviations are used as the texture representation for the image.

3.2. Co-occurrence matrix representation

This approach explores the texture features by analyzing the gray-tone spatial dependencies [16]. We first define a matrix of relative frequencies with which two pixels separated by distance d at a specified angle occur on the image, one with gray-tone i and the other with gray-tone j. Such a matrix depends on the angle selected. We construct four such matrices, corresponding to $0^o, 45^o, 90^o$ and 135^o.

After we construct the co-occurrence matrices, various statistical properties can be extracted as the texture feature. In this paper, we use the most effective two features, i.e. contrast (CTR), and inverse difference moment(IDM).

Since we have 2 features for each orientation, a length-8 vector is used as the texture representation.

3.3. Feature vector

Both the wavelet texture representation and the co-occurrence matrix texture representation consist of multiple components and can be represented as a feature vector:

$$F_i = [f_{i1}, ..., f_{ik}, ..., f_{iN}] \tag{7}$$

N equals 10 in the wavelet case and equals 8 in the co-occurrence matrix case. Note that within the same feature vector, the components f_{ik}'s may be defined over different physical domains. For example, one may be CTR while another may be IDM. Their dynamic ranges may vary drastically.

4. THE INTEGRATED MODEL

The relevance feedback technique described in Section 2 is a powerful technique but it is only applicable in the vector IR model. In order to make use of it, we need to develop techniques which can convert the image feature vectors into weight vectors in vector model. In the remaining of the section, we will describe two approaches. One is base on $tf \times idf$ and the other is based on Gaussian Normalization.

4.1. $tf \times idf$

In text retrieval, the product of tf and idf provides a good estimation of the weights of the terms to a document. Motivated by tf and idf, we propose the factors of *component importance (ci)* and *inverse collection importance (ici)* in image retrieval. The factor of ci captures the relative importance of components within a feature vector while the factor of ici captures the importance of components across different feature vectors over the whole collection.

To estimate ci, note that ci and f_{ik} have quite similar meanings. The former represents the relative important of a component, and the latter represents the magnitude of to what extent a property appears in an image. But to have an accurate estimation, we need to take into account the fact that f_{ik}'s may be defined over different physical domains. To normalize their dynamic ranges to comparable scale, we propose the following to estimate ci:

$$ci_i = [\frac{f_{i1}}{mean_1}, ..., \frac{f_{ik}}{mean_k}, ..., \frac{f_{iN}}{mean_N}]$$

where $mean_k$ is the mean of f_{ik} over all the images.

Just like in the text retrieval, using tf alone may not be a good estimation of the term weights, we also need to estimate the ici factor, which captures the importance of a component across the whole image collection:

$$ici_i = [log_2(\sigma_{i1}+2), ..., log_2(\sigma_{ik}+2), ..., log_2(\sigma_{iN}+2)]$$

where σ_{ik} is the standard deviation of the k^{th} component in ci_i over all the images in the collection. Note that, ici penalizes those components who have small discriminating power but favors those having large discriminating power. This intuition justifies the standard deviation σ being a good measure of ici.

The final weight vector W_i is the product of ci and ici:

$$W_i = ci_i \times ici_i \tag{8}$$

After the conversion from F_i to W_i, the relevance feedback technique described in section 2 can be applied.

4.2. Gaussian Normalization

There are different ways of converting a feature vector to a weight vector. In the previous subsection we described a $tf \times idf$ base approach, in which the weights are incorporated inside the vector itself. The weights can also be outside the vector. In this subsection we will describe an approach which first normalizes each component to the same importance and then dynamically adjusts the weights during the relevance feedback process.

Suppose there are M images in the database. Then we can form an $M \times N$ feature matrix $F = f_{ij}$, where f_{ij} is the jth feature component in feature vector F_i. Now, each column of F is a length-M sequence of the jth feature component, represented as F_j. Our goal is to normalize the entries in each column to the same range so as to ensure that each individual feature component receives equal weight in determining the similarity between two vectors. An effective way of doing this is to use Gaussian Normalization. Assuming the feature sequence F_j to be a Gaussian sequence, we compute the mean m_j and standard deviation σ_j of the sequence. We then normalize the original sequence to a N(0,1) sequence as follows:

Table 1. Retrieval precision
wv: wavelet based; co: co-occurrence matrix based.

	0 rf	1 rf	2 rf	3 rf
wv(tf)	76.87	82.13	84.33	85.50
wv($tf \times idf$)	77.27	82.33	85.13	85.53
wv(Gaussian)	77.67	80.27	80.47	80.53
co(tf)	57.53	63.20	65.13	66.07
co($tf \times idf$)	57.80	63.47	65.13	66.40
co(Gaussian)	44.33	48.53	48.80	48.80

$$f_{ij} = \frac{f_{ij} - m_j}{\sigma_j} \tag{9}$$

It is easy to prove that after the normalization according to (9), the probability of a feature component value being in the range of [-1, 1] is 68%. If we use $3\sigma_j$ in the denominator, according to the 3-σ rule, the probability of a feature component value being in the range of [-1, 1] is approximately 99%. In practice, we can consider all of the feature component values are within the range of [-1,1]. Therefore, this normalization process ensures the equal emphasis of the feature components within a feature vector.

Our final goal is put more emphasis on relevant components and less emphasis on non-relevant components. This can be achieved by using relevance feedback. For all the images that are marked with *relevant* by the user, stack their F_i's to form a $M' \times N$ matrix, where M' is the number of relevant images. In this way, each column of the matrix is a length-M' sequence of F_j. If all the relevant images have similar values for the component j, it means that the component j is relevant to the query. On the other hand, if the values for the component j are very different among the relevant images, then the component j is not relevant to the query. Therefore, the inverse of the standard deviation of the F_j sequence is a good measure of the weight for component j. That is, the smaller the variance, the larger the weight and vice versa.

5. EXPERIMENTAL RESULTS

Our testing image set consists of 384 texture images. The original 24 512×512 texture images are obtained from MIT Media Lab at ftp://whitechapel.media.mit.edu/pub/VisTex/. Each image is then cut into 16 128×128 non-overlap small images. The 16 images from the same big image are considered to be relevant images. Each of the 384 images is selected as the query image and top 15 best matches are returned. The average retrieval precision of 384 query images is summarized in Table 1, where the precision is defined as

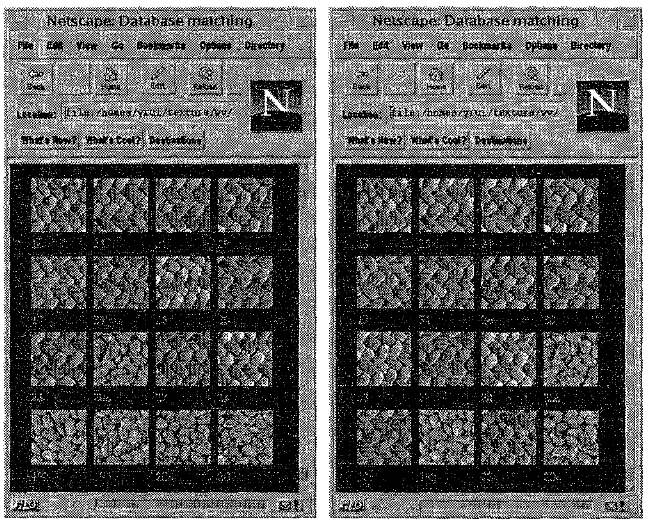
$$precision = \frac{relevant\ \ images}{returned\ \ images} \times 100\% \tag{10}$$

There are four columns in the table. 0 rf stands for no relevance feedback; 1 rf corresponds to 1 iteration of relevance feedback; and so forth.

The purpose of the experiments is not to compare one texture representation vs another, but rather to compare the retrieval performance with relevance feedback vs the performance without relevance feedback. Some observations can be made.

1. The retrieval precision is improved considerably in the feedback case than that in the non-feedback case.
2. The precision increase in the first iteration of feedback is the largest. Subsequent feedbacks will only achieve minor improvement in precision. This is a very desirable property, since this will guarantee that an acceptable retrieval result is achieved within a limited amount of feedback cycles.
3. Using $(tf \times idf)$ to estimate the weights results in a better retrieval performance than using tf alone.
4. Although in the results reported here the $tf \times idf$ approach is always better than the Gaussian Normalization approach, there do exist cases that the latter is better than the former. Furthermore, one major advantage of using the latter is that it is more robust to unknown feature components. That is, if we carefully select the feature components, $tf \times idf$ might do a better job than Gaussian Normalization. But if we do not fine tune the feature components, in most cases, Gaussian Normalization will do a better job.

An example retrieval result for 0 rf and 1 rf is given in Figure 1. A web-based on-line demo of the system is accessible at http://quark.ifp.uiuc.edu:8080.



(a) (b)

Figure 1: (a)Before relevance feedback. (b)After relevance feedback.

6. CONCLUSIONS

Unlike other existing image retrieval systems, where only the IP techniques are explored, the proposal integration approach in this paper explores the techniques in both IP and IR. Specifically, approaches of converting image feature vectors (IP domain) to weighted-term vectors (IR domain) are proposed in this paper. Furthermore, the relevance feedback technique from the IR domain is used in content-based image retrieval to demonstrate the effectiveness of this conversion. Experimental results show that the image retrieval precision increases considerably by using the proposed integration approach.

7. REFERENCES

[1] M. Flickner, H. Sawhney, W. Niblack, and J. Ashley, "Query by image and video content: The qbic system," *IEEE Computer*, 1995.

[2] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. fe Shu, "The virage image search engine: An open framework for image management," in *SPIE Storage and Retrieval for Still Image and Video Databases IV*.

[3] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, 1983.

[4] J. P. Callan, W. B. Croft, and S. M. Harding, "The inquery retrieval system," in *Proc of 3rd Int Conf on Database and Expert System Application*.

[5] T. S. Huang, S. Mehrotra, and K. Ramchandran, "Multimedia analysis and retrieval system (MARS) project," in *Proc of 33rd Annual Clinic on Library Application of Data Processing - Digital Image Access and Retrieval*, 1996.

[6] S. Mehrotra, Y. Rui, K. Chakrabarti, M. Ortega, and T. S. Huang, "Multimedia analysis and retrieval system," in *(submitted to) The 3rd Int. Workshop on Information Retrieval Systems*, 1997.

[7] Y. Rui, A. C. She, and T. S. Huang, "Automated shape segmentation using attraction-based grouping in spatial-color-texture space," in *Proc. IEEE Int. Conf. on Image Proc.*, 1996.

[8] Y. Rui, A. C. She, and T. S. Huang, "Modified fourier descriptors for shape representation – a practical approach," in *Proc of First International Workshop on Image Databases and Multi Media Search*.

[9] Y. Rui, T. S. Huang, S. Mehrotra, and M. Ortega, "Automatic matching tool selection using relevance feedback in MARS," in *Proc. of 2nd Int. Conf. on Visual Information Systems*, 1997.

[10] Y. Rui, T. S. Huang, S. Mehrotra, and M. Ortega, "A relevance feedback architecture in content-based multimedia information retrieval systems," in *Proc of IEEE Workshop on Content-based Access of Image and Video Libraries, in conjunction with IEEE CVPR '97*, 1997.

[11] Y. Rui, T. S. Huang, and S. Mehrotra, "Human perception subjectivity and relevance feedback in multimedia information retrieval," in *(submitted to) Storage and Retrieval of Images/Video Databases VI, EI'98*, 1998.

[12] S. Mehrotra, Y. Rui, M. Ortega-B., and T. S. Huang, "Supporting content-based queries over images in MARS," in *Proc. of IEEE Int. Conf. on Multimedia Computing and Systems*, 1997.

[13] M. Ortega, Y. Rui, K. Chakrabarti, , S. Mehrotra, and T. S. Huang, "Supporting similarity queries in MARS," in *(submitted to)ACM Conf. on Multimedia*, 1997.

[14] C. Buckley and G. Salton, "Optimization of relevance feedback weights," in *Proc. of SIGIR'95*.

[15] W. M. Shaw, "Term-relevance computations and perfect retrieval performance," *Information Processing and Management*.

[16] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Texture features for image classification," *IEEE Trans. on Sys, Man, and Cyb*, vol. SMC-3, no. 6, 1973.