A Stochastic Model for Content-Based Image Retrieval

Mei-Ling Shyu[1], Shu-Ching Chen[2], Lin Luo[2], and Chi-Min Shu[3]

[1] Department of Electrical and Computer Engineering, University of Miami,
Coral Gables, FL USA
shyu@miami.edu
[2] Distributed Multimedia Information System Laboratory
School of Computer Science, Florida International University
Miami, FL USA
{chens, lluo0001}@cs.fiu.edu
[3] Department of Environmental and Safety Engineering
National Yunlin University of Science and Technology
Touliu, Yunlin Taiwan, R.O.C.
shucm@yuntech.edu.tw

Abstract. Multimedia data, typically image data, is increasing rapidly across the Internet and elsewhere. To keep pace with the increasing volumes of image information, new techniques need to be investigated to retrieve images intelligently and efficiently. Content-based image retrieval is always a challenging task. In this paper, a stochastic model, called Markov Model Mediator (MMM) mechanism, is used to model the searching and retrieval process for content-based image retrieval. Different from the common methods, our stochastic model carries out the searching and similarity computing process dynamically, taking into consideration not only the image content features but also other characteristics of images such as their access frequencies and access patterns. Experimental results demonstrate that the MMM mechanism together with the stochastic process can assist in retrieving more accurate results for user queries.

1 Introduction

Recently, the volumes of multimedia information are growing rapidly, and it becomes easier to access multimedia data due to the popularity of the Internet. There is a great need for efficient image retrieving methods. Content-Based Image Retrieval (CBIR) is an active research area where the image retrieval queries are based on the content of multimedia data. A lot of research work has been done, which resulted in a number of systems and techniques in both the academic and commercial domains. For example, the QBIC system [2] and Virage's VIR engine [12] are two most notable commercial image retrieval systems, while VisualSEEk [11] and PhotoBook [6] are well-known academic image retrieval systems.

The objective of a CBIR system is to enable the user to efficiently find and retrieve those images he/she wants from the database. Most of the existing CBIR systems retrieve images in the following manner. First, they built the indexes based on the low-level features such as color, texture and shape of the images in the database. The index of a query image is also generated when the query is issued. Secondly, they searched through the whole database and measured the similarity of each image to the query. Finally, the results were returned in a sorted order of the similarity matching level.

In this paper, the Markov model mediator (MMM) mechanism that adopts the Markov model framework and the mediator concept [8, 9] is proposed for content-based image retrieval. Markov model is one of the most powerful tools available to the scientists and engineers for analyzing complicated systems. Some research works have been done to integrate the Markov model into the field of image retrieval. Lin et al. [4] used a Markov model to combine the spatial and color information. The hidden Markov model (HMM) was used to parse video data in [14]. In [5], the HMM was employed to model the time series of the feature vector for the cases of events and objects in their probabilistic framework for semantic level indexing and retrieval.

The uniqueness of our model lies in the integration of two relationships when determine the similarity: 1) the relationship between the query and the candidate image; 2) the relationships among all images in the database. A stochastic process that takes into account the image content features and other characteristics of the images is also proposed. Several experiments have been conducted and the experimental results demonstrate that the MMM mechanism together with the stochastic process can assist in retrieving more accurate results for user queries.

The remainder of this paper is organized as follows. Section 2 reviews the key components of the MMM mechanism and introduces the stochastic process for information retrieval. Section 3 presents our experiments and discusses the experimental results. Conclusion and future work are given in Section 4.

2 The Stochastic Model

2.1 Markov Model Mediator (MMM) Mechanism

Markov model mediator, for short, MMM, is a probabilistic-based mechanism that adopts the Markov model framework and the mediator concept. A Markov model is a well-researched mathematical construct that is powerful in analyzing complicated systems [3, 7]; while a mediator can collect and report information from time to time [13].

Definition 1: A MMM is represented by a 5-tuple $\lambda = (\mathcal{S}, \mathcal{F}, \mathcal{A}, \mathcal{B}, \Pi)$, where $\mathcal{S}$ is a set of images called states; $\mathcal{F}$ is a set of features; $\mathcal{A}$ denotes the states transition probability distribution; $\mathcal{B}$ is the observation symbol probability distribution; and Π is the initial state probability distribution.

The elements in $\mathcal{S}$ and $\mathcal{F}$ determine the dimensions of $\mathcal{A}$ and $\mathcal{B}$. If there are totally s images in $\mathcal{S}$ and the number of distinct features in $\mathcal{F}$ is f, then the dimensions of $\mathcal{A}$ is $s \times s$ and $\mathcal{B}$ is $s \times f$. The relationships of the images are modeled by the sequences of the MMM states connected by transitions, i.e., each entry (m, n) in $\mathcal{A}$ indicates the relationship between images m and n. A training data set consisting of the access patterns and access frequencies of the queries issued to the database is used to train the model parameters for a MMM.

2.2 Formulation of the Model Parameters

Each MMM has three important probability distributions: $\mathcal{A}$, $\mathcal{B}$, and Π. These distributions are critical for the stochastic process and can be obtained from the training data set.

Definition 2: For the images in database d and their corresponding content features, the training data set consists of the following information:

- A set of queries $Q = \{q_1, q_2, \cdots, q_q\}$ that are issued to the database in a period of time;
- The usage patterns $use_{m,k}$ and access frequencies $access_k$ of the queries.
  - $use_{m,k} = 1$ means that image m is accessed by q_k and 0 otherwise.
  - $access_k$ denotes the access frequency of query q_k per time period.

Definition 3: The relative affinity measurements indicate how frequently two images are accessed together, and is defined as follows.

$$aff_{m,n} = \sum_{k=1}^{q} use_{m,k} \times use_{n,k} \times access_k \tag{1}$$

Based on the relative affinity measurements obtained from Equation 1, the state transition probability distribution $\mathcal{A}$ is constructed as follows.

$$a_{m,n} = \frac{f_{m,n}}{f_m} \tag{2}$$

where

$$f_{m,n} = \frac{aff_{m,n}}{\sum_{m \in d} \sum_{n \in d} aff_{m,n}} \tag{3}$$

$$f_m = \sum_{n} f_{m,n} \tag{4}$$

Here, $f_{m,n}$ is defined as the joint probability that refers to the fraction of the relative affinity of images m and n in database d with respect to the total relative affinity for all the images in d, and f_m is the marginal probability. $a_{m,n}$ is the conditional probability that refers to the state transition probability for a MMM, where $a_{m,n}$ is the element in the $(m, n)^{th}$ entry in $\mathcal{A}$.

The observation symbol probability $\mathcal{B}$ denotes the probability of observing an output symbol from a state, where the observed output symbols representing the distinct features of the images and the states representing the images in the databases. A temporary matrix ($\mathcal{BB}$) is defined to capture the appearance of features in the images, whose rows are all the distinct images and columns are all the distinct features. The value in the $(p,q)^{th}$ entry is 1 if feature q appears in image p, and 0 otherwise. Then the observation symbol probability distribution $\mathcal{B}$ can be obtained via normalizing $\mathcal{BB}$ per row.

The initial state probability distribution Π indicates the preference of the initial states for queries. For any image $m \in d$, its initial state probability is defined as:

$$\Pi = \{\pi_m\} = \frac{\sum_{k=1}^{q} use_{m,k}}{\sum_{l \in d} \sum_{k=1}^{q} use_{l,k}} \tag{5}$$

2.3 Stochastic Process for Information Retrieval

The desired images are captured through a dynamic programming algorithm that calculates the current edge weights and cumulative weights via a stochastic process. Assume there are N images in the databases, and each query is denoted as $q_k = \{o_1, o_2, \cdots, o_T\}$, where T is the total number of features appeared in the query.

Definition 4: $W_t(i,j)$ is defined as the edge weight of the edge $S_i \rightarrow S_j$ on evaluating the t^{th} feature (o_t) in the query, where $1 \leq i,j \leq N$, $1 \leq t \leq T$, $S_i \in \mathcal{S}$ and $S_j \in \mathcal{S}$.

Definition 5: $D_t(i,j)$ is defined as the cumulative edge weight of the edge $S_i \rightarrow S_j$ on evaluating the t^{th} feature (o_t) in the query, where $1 \leq i,j \leq N$, $1 \leq t \leq T$, $S_i \in \mathcal{S}$ and $S_j \in \mathcal{S}$.

Based on Definitions 4 and 5, the dynamic programming algorithm is given as follows. At $t = 1$,

$$W_1(i,j) = \begin{cases} \pi_{S_i} b_{S_i}(o_1) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

$$D_1(i,j) = W_1(i,j) \tag{7}$$

For $1 \leq t \leq T-1$, the values of $W_{t+1}(i,j)$ and $D_{t+1}(i,j)$ are calculated using the values of $W_t(i,j)$ and $D_t(i,j)$.

$$W_{t+1}(i,j) = \max_k (D_t(k,i)) a_{S_i,S_j} b_{S_j}(o_{t+1}) \tag{8}$$

$$D_{t+1}(i,j) = (\max_k D_t(k,i)) + W_{t+1}(i,j) \tag{9}$$

where a_{S_i,S_j}, $b_{S_j}(o_k)$ and π_{S_i} are elements in probability distributions $\mathcal{A}$, $\mathcal{B}$ and Π, respectively.

Following are the steps for image retrieval using the dynamic programming algorithm in our proposed stochastic model.

1. For the first feature o_1, calculate $W_1(i,j)$ and $D_1(i,j)$ by Equations 6 and 7.
2. For the rest of the features o_t where $2 \leq t \leq T$, calculate $W_t(i,j)$ and $D_t(i,j)$ according to Equations 8 and 9.
3. Sum up each column in $W_t(i,j)$ and $D_t(i,j)$. That is, calculate $sumW_t(j) = \sum_i W_t(i,j)$ and $sumD_t(j) = \sum_i D_t(i,j)$.
4. Rank the images to the user query based on their corresponding values in $sumD_T(j), \ldots, sumD_1(j)$.
   - First, an image is ranked according to its value in $sumD_T(j)$.
   - If two or more images have the same values, then $sumD_{T-1}(j)$ for these images are compared.

3 Experiments

A testbed of 400 color images with various dimensions is used as the image database for the experiments. One MMM model will be constructed for this image database. The MMM model has three model parameters: $\mathcal{A}$, $\mathcal{B}$, and Π probability distributions. A training data set that consists of the query usage patterns and access frequencies together with the images in the database are required for constructing these model parameters for the MMM model. $\mathcal{A}$ can be obtained according to Equations 1 to 4; $\mathcal{B}$ can be converted from $\mathcal{BB}$; and finally Π can be determined by using Equation 5. In our experiments, we do not use the query-by-example strategy for the queries. However, the query-by-example strategy can be easily implemented and employed in our proposed model.

3.1 Construction of the Model

In our experiments, there are eight typical queries issued to the image database with the same access frequency. Each query accesses one or more features from one or more images in the database. In order to support the semantic level queries, both color information and object location information of the images are exploited as the features of the images for the construction of $\mathcal{B}$.

In our experiments, each image has a 21-bins feature vector: twelve for color descriptions and nine for object location descriptions. Color information is the image's color histogram algorithm in the HSI color space. The whole color space is divided into twelve areas according to combinations of different ranges of hue, saturation and intensity values. These areas are: black, white, red, red-yellow (ry), yellow, yellow-green (yg), green, green-blue (gb), blue, blue-purple (bp), purple, and purple-red (pr). Colors with pixel number less than 5% are ignored. As for the object information of an image, the SPCPE (Simultaneous Partition and Class Parameter Estimation) algorithm proposed in [10, 1] is used to extract the object information. Each image is divided into 3×3 regular regions, ordered from left to right and top to bottom as: L1, L2, L3, L4, L5, L6, L7, L8 and L9. The locations of the objects within an image are represented by the locations of their centroids. If there is an object whose centroid falls into the associated

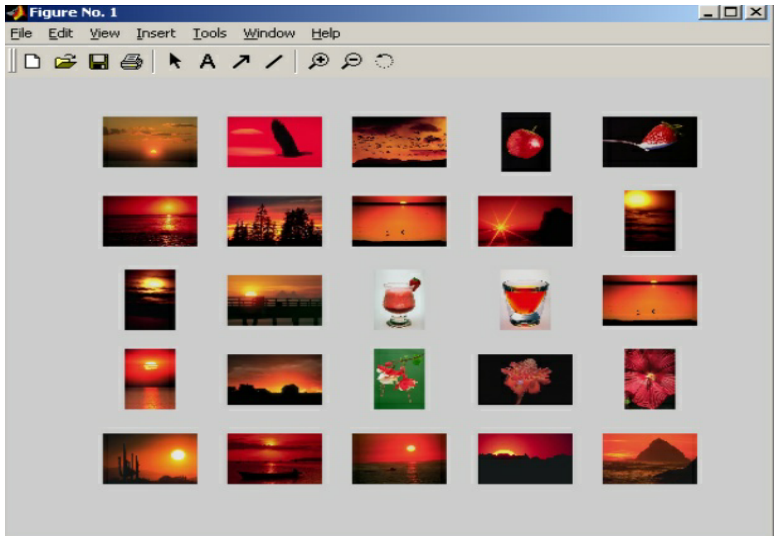

Fig. 1. Snapshot of the query with "red" feature

region, the value 1 is assigned to that location element in the feature vector and the value is 0 otherwise.

Once a query is issued, the stochastic process with the proposed dynamic programming algorithm in Section 2.3 is carried on to retrieve the candidate images that matched the query. The similarity matching degrees of the images with respect to the query are estimated based on the steps described in Section 2.3. The candidate images are sorted decreasingly in accordance with their matching degrees.

3.2 Experiment Results

Experiments have been conducted by issuing various types of queries to the system. We use the snapshot of the query result to illustrate how our proposed model can obtain more accurate retrieval results with respect to a query. In the snapshots, the candidate images are shown with an descending order from *top left* to *bottom right*.

• ***Query Type I:*** *the number of features in the query = 1.*
In this type of queries, the user specifies only one feature of the desired images, either the color information or the object location information. The system retrieves those images that have this specified feature.

Example: Querying with the "red" color feature.
In this query example, the user wants to retrieve the images that have the red color. Figure 1 shows the snapshot of the window containing the top 25 can-

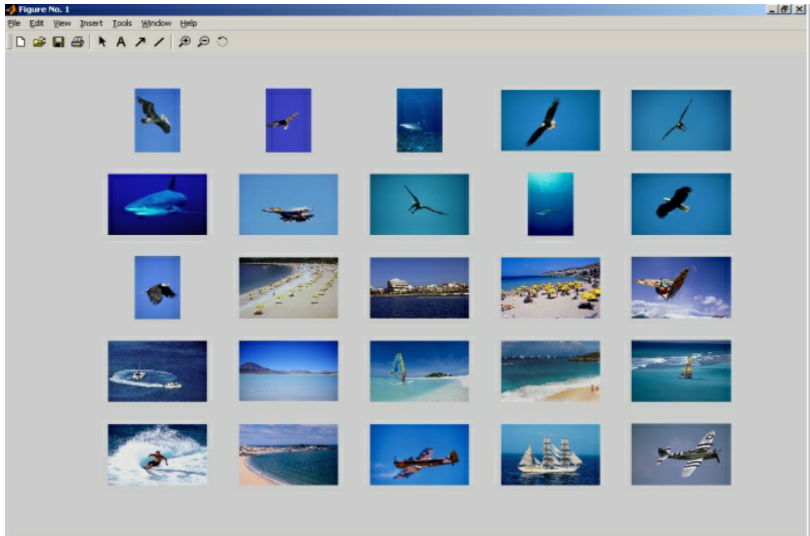

Fig. 2. Snapshot of the query with "white","blue" and "L5" features

didate images. There are quite a few images in our image collection containing the "*red*" color feature. As can be seen from this figure, all the retrieved images have "red" color in them.

• ***Query Type II:*** *the number of features in the query* ≥ 2.
This type of queries specifies two or more features including the color features and/or the object location features in the queries.

Example: Querying with the "white", "blue", and "L5" features.
This example shows a three-feature query, including two color features (white and blue) and one object location feature (L5). The snapshot for this query is shown in Figure 2. As can be seen from this figure, all these top 25 images have the "white" and "blue" colors, and have one or more objects at location "L5" within the image, which are the desired features in the query.

4 Conclusion and Future Work

In this paper, the Markov Model Mediator (MMM) mechanism is applied to content-based image retrieval. A stochastic approach based on MMM is used to traverse the database and retrieve the images satisfying the query. This approach performs similarity comparison based on not only the relationship between the query image and the target image, but also the relationships among all the images within the database. Experiments with different numbers of features in the queries were conducted to illustrate how our proposed stochastic-based model works for various types of queries. The experimental results demonstrated that our model can obtain more accurate retrieval results.

The time complexity of the current dynamic programming algorithm is $O(n^2)$. To achieve less complexity, we would like to try to build the dynamic programming algorithm based on the forward variable introduced in the hidden Markov model (HMM) since it has the potential to reduce the time complexity to $O(n)$.

Acknowledgment

For Shu-Ching Chen, this research was supported in part by NSF CDA-9711582.

References

1. Chen, S.-C., Sista, S., Shyu, M.-L., Kashyap, R.L.: An Indexing and Searching Structure for Multimedia Database Systems. IS&T/SPIE Conference on Storage and Retrieval for Media Databases 2000, (2000) 262-270.
2. Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., Yanker, P.: Query by Image and Video Content: The QBIC System. IEEE Computer, 28(9) (1995) 23-31.
3. Frank, O., Strauss, D.: Markov Graphs. Journal of the American Statistical Association, 81 (1986) 832-842.
4. Lin, H.C., Wang, L.L., Yang, S.N.: Color Image Retrieval Based on Hidden Markov Models. IEEE Transactions on Image Processing, 6(2) (1997) 332-339.
5. Naphade, M.R., Huang, T.S.: A Probabilistic Framework for Semantic Indexing and Retrieval in Video. IEEE Transactions on Multimedia, 3(1) (2001).
6. Pentland, A., Picard, R.W., Sclaroff, S.: Photobook: Tools for Content-based Manipulation of Image Databases. Proc. Storage and Retrieval for Image and Video Databases II, Vol. 2185, SPIE, Bellingham, Washington (1994) 34-47.
7. Rabiner, L.R., Huang, B.H.: An Introduction to Hidden Markov Models. IEEE ASSP Magazine, 3(1) (1986) 4-16.
8. Shyu, M.-L., Chen, S.-C., Kashyap, R.L.: A Probabilistic-based Mechanism for Video Database Management Systems. IEEE International Conference on Multimedia and Expo (ICME2000), New York (2000) 467-470.
9. Shyu, M.-L., Chen, S.-C., Shu, C.-M.: Affinity-based Probabilistic Reasoning and Document Clustering on the WWW. the 24th IEEE Computer Society International Computer Software and Applications Conference (COMPSAC), Taipei, Taiwan (2000) 149-154.
10. Sista, S., Kashyap, R.L.: Unsupervised Video Segmentation and Object Tracking. IEEE International Conference on Image Processing, Japan (l999).
11. Smith, J.R., Chang, S.F.: VisualSEEK: A Fully Automated Content-based Image Query System. In Proceedings ACM Intern. Conf. Multimedia, Boston (1996) 87-98.
12. http://www.virage.com
13. Wiederhold, G.: Mediators in the Architecture of Future Information Systems. IEEE Computers, (1992) 38-49.
14. Wolf, W.: Hidden Markov Model Parsing of Video Programs. Presented at the International Conference of Acoustics, Speech and Signal Processing, (1997).