2007

Content-Based Image Retrieval Using Associative Memories

Arun D. Kulkarni
University of Texas at Tyler, akulkarni@uttyler.edu

Content-Based Image Retrieval Using Associative Memories

Arun Kulkarni
Computer Science Department
The University of Texas at Tyler, Tyler, TX 75799
Arun_Kulkarni@UTTyler.edu

ABSTARCT

The rapid growth in the number of large-scale repositories has brought the need for efficient and effective content-based image retrieval (CBIR) systems. The state of the art in the CBIR systems is to search images in database that are "close" to the query image using some similarity measure. The current CBIR systems capture image features that represent properties such as color, texture, and/or shape of the objects in the query image and try to retrieve images from the database with similar features. In this paper, we propose a new architecture for a CBIR system. We try to mimic the human memory. We use generalized bi-directional associative memory (BAMg) to store and retrieve images from the database. We store and retrieve images based on association. We present three topologies of the generalized bi-directional associative memory that are similar to the local area network topologies: the bus, ring, and tree. We have developed software to implement the CBIR system. As an illustration, we have considered three sets of images. The results of our simulation are presented in the paper.

Keywords: Content based image retrieval, Multimedia Database, and Associative Memory.

1. INTRODUCTION

Content-based image retrieval (CBIR) today is an active area of research. The rapid growth in the number of large-scale image repositories in many domains such as medical image management, multimedia libraries, document archives, art collection, geographical information the systems, law enforcement management, environmental monitoring, biometrics, and journalism has bought the need for efficient CBIR mechanisms. Many ideas from fields such as computer vision, database, image processing, and information retrieval are used in CBIR [11]. There are three broad categories of users that browse through a large set of images from unspecified sources. Users in the first category search a target image by association. The result of the source can be manipulated interactively by relevance feedback. Users in the second category try to search a target image, which may be a precise copy of the image in the mind or may be another image of the same object for which the user has an image. Users in the third category aim at retrieving an image of a specific class. The image class may be derived from labels or emerge from the database. CBIR systems commonly use a set of features for image representation in addition to some meta-information that may have been stored in key words. Most systems use color histograms to compare images. The ability of retrieve images when color features are similar across the database is achieved by using texture features. Shape is also an important attribute that is employed in computing similarity of regions in images [7]. CBIR is a very active field of research. Many papers have been published in the last few years in this area. However, most papers deal with searching images from the database using some similarity measure. Many similarity measures have been proposed in the literature [10]. In this paper, we propose a new approach for CBIR. We try to mimic the human brain. Association is one of the fundamental characteristics of the human brain. The human memory operates in an associative manner; that is, a portion of recollection can produce an associated stream of data from the memory. The human memory can retrieve a full image from a partial or noisy version of the image as the query image. Furthermore, given a query image as the input, the human brain can recall associated images that have been stored in the past. The human memory can respond to abstract queries. The main disadvantage of the current CBIR systems is their inability to respond to abstract queries. Abstract queries are based on a notion of similarity, the concept that is difficult to capture in a mathematical model. For example, if we see an image of a person, we can recall images of his house, spouse, and/or his car. The associative storage and retrieval mechanism is not explored fully in present CBIR systems. We propose a new architecture for CBIR that is based on generalized associative memories. The rest of the paper is organized as follows: In Section 2, we provide overview of CBIR systems. In Section 3, we describe the architecture of our proposed CBIR system, and introduce generalized bi-directional associative memory models. In Section 4, we present results.

2. BACKGROUND

Stages in a typical CBIR system include annotation, preprocessing, and feature extraction. In order to store an image it is first annotated. The preprocessing stage deals with geometric and radiometric correction, mapping the image from red, blue, green (RBG) color space to hue, saturation, and intensity (HIS) color space, and similar such operations. In the feature extraction stage, features based on attributes such as the color, texture, and/or shape are extracted. Most CBIR systems use color histograms to compare the query image with images that are stored in the database. Often color histograms alone are not sufficient to retrieve desired images from the database, because a color histogram may represent multiple images in the database. There are many CBIR systems that use texture and shape features in addition to color features [3, 4]. Spatial relationship between the objects in the image can also be used to describe contents of the image. Freeman [1] defines eleven primitive spatial relations between two objects in the image. After capturing information from the query image and the image in the database into features vectors, elements of the feature vectors are compared based on the similarity function. The most commonly used similarity function is the Minkowski distance that is given by Equation (1).

$$d\left(\mathbf{f}^{q},\mathbf{f}^{d}\right)=\left[\sum_{j=1}^{n}\left|f_j^{q}-f_j^{d}\right|^2\right]^{\frac{1}{2}} \qquad (1)$$

Where $\mathbf{f}^{q}$ and $\mathbf{f}^{d}$ represent the features vectors corresponding to the query image and the image in the database, and n is the number of features. The query image is compared with images in the database and the images in the database are ranked based on the similarity measure. Images that are similar to the query image are retrieved and displayed. While it is feasible to retrieve a desired image from a small collection by exhaustive search, more effective techniques are needed with a larger database. The well known indexing technique is used for efficient retrieval. Other features of a typical CBIR system include defining query feature space and displaying query results. The first component of the query space is the selection of a subset of images from the large image archive. In order to reduce the query space images in the database are clustered into a small number of categories. In this paper, we propose a new architecture for a CBIR system. Our architecture is based on associative memory. In the following section we describe the proposed architecture and structures of the generalized bi-directional associative memory.

3. METHODOLOGY

A schematic block diagram of the proposed CBIR system is shown in Figure 1. There are two data paths in a CBIR system. One deals with storing a set of images in the database, while other path deals with a query image. We group images in the database in predefined categories. It is also possible to use clustering to group images. All images are stored in I-file, and the corresponding feature vectors are stored in F-files. We maintain an index table that defines feature vector to image mapping. The most important feature of the proposed CBIR system is that we use associative memory for storage and retrieval of feature vectors. The basic functions of an associative memory are to store associative pairs through a self-organizing process and to produce an appropriate response pattern on receipt of the associated stimulus input pattern. The stimulus and the response pattern may represent features of corresponding images. Several associative memory models have been proposed to simulate human memory. However, the potential of these models have not been explored for CBIR. Linear associative memories have been studied extensively by Kohonen [5]. Bidirectional associative memories (BAMs) have studied by Kosko [6]. An associative network is designed to map stimulus vectors $\left\{\mathbf{x}_1,\mathbf{x}_2,\ldots\mathbf{x}_n\right\}$ to response vectors $\left\{\mathbf{y}_1,\mathbf{y}_2,\ldots\mathbf{y}_n\right\}$. In an auto-associative network, the response vector $\mathbf{y}_i$ and the corresponding stimulus vector $\mathbf{x}_i$ are the same. In a hetero-associative memory, the stimulus and the response vector are not identical. Associative memories are often able to produce correct response patterns even though

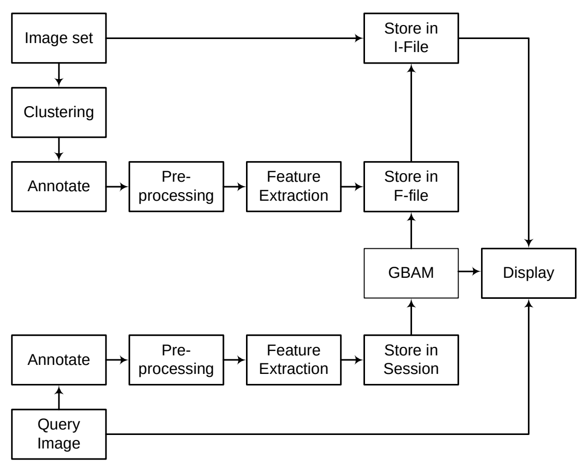

Figure 1. Schematic block diagram of the CBIR system

stimulus patterns are distorted or incomplete. Conventional BAMs are used to store are retrieve pairs of stimulus and response patterns. However, if the number of associated inputs and/or outputs is more than two; that is, instead of pairs of vectors, if we want to store triplets or quadruplets of vectors, then we need to use generalized BAMs.

The discrete BAM is the logical extension of autocorrelator. A three-layer network with feedback that simulates a BAM is shown Figure 2. Pairs of vectors

$(\mathbf{x}_i, \mathbf{y}_i)$ can be stored with the BAM by summing bipolar correlation matrices. In a $m \times n$ BAM, *n* neurons in layer L1 represent the bottom-up field $F_x = \{x_1, x_2, \ldots x_n\}$, and *m* neurons in layer L2 by the top-down field $F_y = \{y_1, y_2, \ldots y_n\}$. The two fields are interconnected by a $m \times n$ synoptic weight matrix **W**. The neuron states in field F_x and field F_y are the units of the short-term memory (STM). The connection matrix **W** is the unit of the long-term memory (LTM). Information passes forward from one layer to another through the connection matrix **W.** Information passes backward through the matrix transpose $\mathbf{W}^T$ [8, 9].

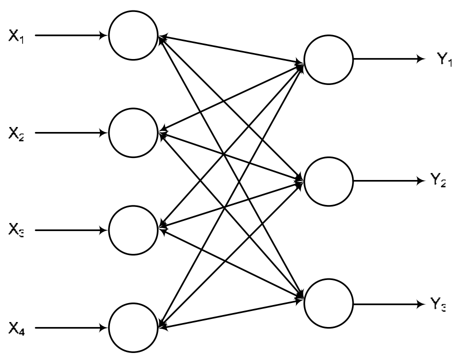

Figure 2. Bi-directional associative memory

Let for $i = 1, 2, \ldots N$ be the *N* pairs of patterns to be encoded in a BAM. One way to memorize the association $(\mathbf{x}_i, \mathbf{y}_i)$ is by forming a correlation matrix $\mathbf{y}_i \mathbf{x}_i^T$. A number of associations can be stored by adding corresponding correlation matrices.

$$\mathbf{W} = \sum_{i=1}^{N} \mathbf{y}_i \mathbf{x}_i^T \qquad (2)$$

In order to recall vector $\mathbf{y}_i$, we can use the stimulus vector $\mathbf{x}_i$ as the input vector. If input pattern vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$ are orthonormal i.e.

$$\mathbf{x}_i \mathbf{x}_j = \begin{cases} 1 \text{ for } i = j \\ 0 \text{ for } i \neq j \end{cases} \qquad (3)$$

then the recall is perfect. If the input vectors are not orthonormal, then the output vector may contain cross talk. In a dual BAM, feedback is achieved with $\mathbf{W}^T$ and is given by

$$\mathbf{W}^T = \sum_{i=1}^{N} \left(\mathbf{y}_i \mathbf{x}_i^T \right)^T = \sum_{i=1}^{N} \mathbf{x}_i \mathbf{y}_i^T \qquad (4)$$

If we assume a nonlinear transfer function for neurons in the BAM, then the recalled output is a nonlinear function of a transformed input vector and is given by

$$\mathbf{y}_i = F(\mathbf{W}\mathbf{x}_i) \qquad (5)$$

With the feedback the input vector $\mathbf{x}_i$ can be estimated as

$$\mathbf{x}_j = F(\mathbf{W}^T \mathbf{y}_i) \qquad (6)$$

The simplest transfer function for the BAM is a step function. The stable reverberation corresponds to the system energy local minimum. When the BAM neurons are activated, the network quickly evolves to a stable state of two-pattern reverberation or a non-adaptive resonance. In order to improve recall accuracy, the output vector $\mathbf{y}_i$ can be synchronously fed back. The back-and-forth flow of distributed information quickly resonates on a fixed data pair. The sequence can be represented by

$$\begin{aligned}
&\mathbf{x}_i(0) \rightarrow \mathbf{W} \rightarrow \mathbf{y}_i(0) \\
&\mathbf{y}_i(0) \rightarrow \mathbf{W}^T \rightarrow \mathbf{x}_i(1) \\
&\mathbf{x}_i(1) \rightarrow \mathbf{W} \rightarrow \mathbf{y}_i(1) \\
&\mathbf{y}_i(1) \rightarrow \mathbf{W}^T \rightarrow \mathbf{x}_i(2) \qquad (7) \\
&. \qquad . \qquad . \\
&\mathbf{x}_i(n) \rightarrow \mathbf{W} \rightarrow \mathbf{y}_i(n) \\
&\mathbf{y}_i(n) \rightarrow \mathbf{W}^T \rightarrow \mathbf{x}_i(n+1)
\end{aligned}$$

Humpert [2] has suggested generalization of the BAM that can store associated multiple input/output patterns. In this paper, we discuss various models of generalized BAM and illustrate how these models can be used for CBIR. There is a striking similarity between the field topologies of a generalized BAM and topologies for local area networks (LANs). We have developed models of generalized BAM using topologies such as the star, ring, and tree. The generalization of the BAM to several vector fields raises questions regarding the updating process. In a BAM, all units in a field are synchronously updated. By contrast, the sequence of updating weights in a generalized BAM is not obvious. The generalization of a BAM to several fields also raises question of interconnections. The generalized BAMs are extremely useful in content-based image storage and retrieval for multimedia applications. The bus and tree topologies are suitable for storing and retrieving temporal images or image sequences. Also, one needs to consider the capacity of the generalized BAM.

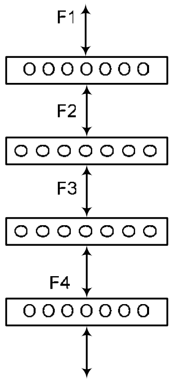

Figure 3. Generalized BAM-Bus Structure

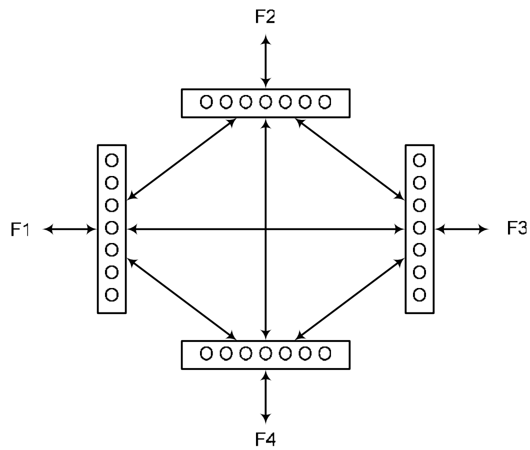

Figure 4. Generalized BAM- Ring Structure

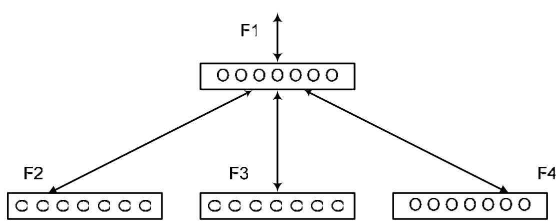

Figure 5. Generalized BAM- Tree Structure

4. RESULTS AND DISCUSSIONS

We have developed software to store and retrieve feature vectors. In our simulation, we have considered three sets of test images of characters {1, 2, 3}, {A, B, C}, and {α, β, γ}. Each character was represented by a 12x12 matrix. Each image was represented by a feature vector of 144 elements. These feature vectors are stored in the generalized BAM. During retrieval, any one image (partial or noisy) from any set was used as the query image, and the corresponding images from the other sets were retrieved. The results are shown in Figures 6 and 7. The first row shows the query images and subsequent rows show corresponding retrieved images. In the present simulation, we have used binary images. However, the method is generic and we can process color images in the similar manner. The generalized BAM stores and recalls feature vectors. We can use feature vectors that may represent color, texture and/or shape of objects in the image. We have presented three possible topologies of BAMg. In present simulation, we used a tree topology for storing three sets of images. The first set contained images of numbers, the second set contained images of characters, and the third set contained images of Greek characters. The bus and tree structures are useful in storing image sequences or temporal images.

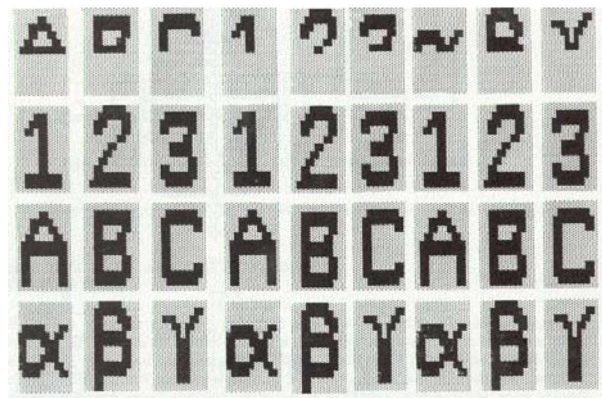

Figure 6. Partial input patterns and the corresponding recalled patterns

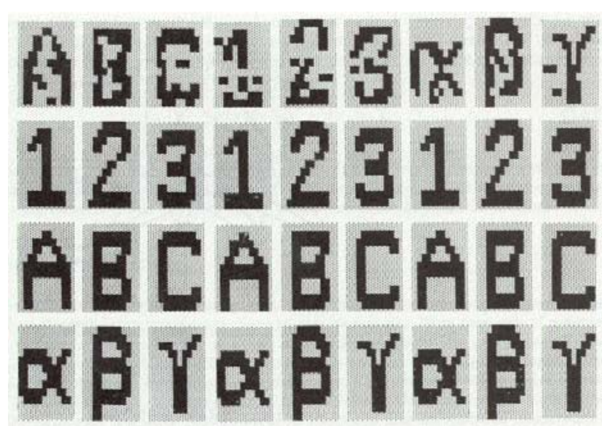

Figure 7. Noise input patterns and the corresponding recalled patterns

5. REFERENCES

[1] J. Freeman, “The modeling of spatial relations”, **Computer Graphics and Image Processing**, vol. 4, 1975, pp 156-171.

[2] B. Humpert, “Bi-directional associative memory with several patterns”, **Proceedings of International Joint Conference on Neural Networks, San Diego**, vol. I, 1990, pp. 741-750.

[3] A. K. Jain and A. Vailaya, “Image retrieval using color and shape”, **Pattern Recognition,** vol. 29, no. 8, 1996, pp 1233-1244.

[4] A. K. Jain and A. Vailaya, “Shape-based retrieval: A case study with trademark image databases”, **Pattern Recognition,** vol. 31, no. 9, 1998, pp 1369-1390.

[5] T. Kohonen, **Self-Organization and Associative Memory.** Springer-Verlag, Berlin, 1988.

[6] B. Kosko, “Bi-directional associative memories”, **IEEE Transactions on Systems, Man and Cybernetic***s* vol. 18, 1988, pp 49-60.

[7] R. Krishnapuram, S. Medasani, S. Jung,, Y. Choi, and R. Balasubramanian, “Content-based image retrieval based on a fuzzy approach”, **IEEE Transactions on Knowledge Engineering**, vol. 16, no. 10, 2004, pp 1185-1199.

[8] A. D. Kulkarni and I. Yazdanpanahi, “Generalized bi-directional memories for image processing”, **Proceedings of SPIE vol. 1826 Intelligent Robots and Computer Vision XI,** 1992, pp 152-159.

[9] A. D. Kulkarni, **Computer Vision and Fuzzy-Neural Systems**. Prentice Hall, Upper Saddle River, NJ, 2001.

[10] S. Santini and R. Jain, R., “Similarity measures” **IEEE Transactions on Pattern Analysis and Machine Intelligenc***e*, vol. 21, no. 9, 1999, pp 871-883.

[11] A. W. M. Smeulders, M. Worring, S. Santini,, A. Gupta, and R. Jain, “Content-based image retrieval at the end of the early years”, **IEEE Transactions on Pattern Recognition and Machine Learning,** vol. 22, no. 12, 2000, pp 1349-1380.