VBIRS-Visual based Image Retrieval System for Generic Web Image Database

Umesh K K
Research Scholar
Department of Studies in Computer Science
University of Mysore, Mysore, India

Suresha
Research Supervisor
Department of Studies in Computer Science
University of Mysore, Mysore, India

ABSTRACT

In this paper, we discussed Visual Based Image Retrieval System to retrieve set of relevant images for the given input image from the large generic image database. We proposed HSV color space model and Haar transform to extract color and texture features. The images are transformed into set of features. These features are used as inputs in Self Organizing Maps (SOM) to train the network for generate the code word. The advantage of SOM is able to preserve topology structure. The cosine similarity measure is used to retrieve similar images with new representation. The experimental results are evaluated over a collection of 10,000 general purpose images to demonstrate the effectiveness of the proposed system.

Keywords

Content-based image retrieval, feature extraction, Image databases, Neural networks, Self-Organizing Map, Similarity measures.

1. INTRODUCTION

Due to rapid changes in digital technologies, digital information such as text, image, video, audio etc., has become popular in recent years. It requires effective indexing and searching tools for large image database. Most of the researcher's have been involved to develop the system to retrieve the set of similar images for the given input image. The content of an image have been used to represent semantically- meaningful on the image database. The derived image features are used to retrieving relevant images semantically from large image database.

In the past few years, many researchers have been involved in the area of Content-Based Image Retrieval (CBIR) system to develop techniques to retrieve unannotated images [1]. These days many people use a Digital image and video libraries are the main source of visual information. Hence it is an open challenge for the research community, to develop cost effective technologies for retrieving, managing and browsing the content in the large image database are still open issues.

Many projects have been started in recent years to develop efficient CBIR systems. The first well known CBIR system is Query by Image Content (QBIC) [2] was developed at the IBM Almaden Research Center. Other systems include MIT's Photobook [3] and its recent version, FourEyes [4], the search engine family of VisualSEEk, WebSEEk, and MetaSEEk [5],[6],[7], which all are developed at Columbia University. The virage [8] is a commercial content-based search engine developed at Virage Technologies.

The Visual Based Image Retrieval System (VBIRS) proposes a Self-Organizing Map (SOM) clustering algorithm to generate the codebook. The codebook is used to compute the signature of every image in the image database [9] [10]. The corresponding signature and its links are stored in the image database. The main advantage of SOM is uses its topology preserving structure. Basically SOM is an unsupervised and iterative clustering algorithm. It finds the optimal set of cluster prototypes based on a grid of artificial neurons whose weights are adapted to match the input vectors in a training set [11]. In the retrieval phase, the distance of the query image and the set of images in the image database are to evaluate based on similarity measure. The set of retrieved similar images and its corresponding links of the images are display at the user interface.

The organization of this paper is as follows: In section 2, the Methods are discussed. The Section 3 describes the experimental results. In section 4, discussed the conclusion

2. METHODS

This section presents a brief description of the data used in our study, the feature extraction, SOM approach and feature representation for image retrieval.

2.1 Materials

The low resolution Web-crawled miscellaneous database has been used in this study (http://wang.ist.psu.edu/docs/related.shtml) [16]. This image database has a collection of 100 subjects contains 10,000 low resolution web-crawled miscellaneous database with the size of 128 x 85 and 85 x 128 resolutions. Most of the images are color photographs in JPEG format. We analysed that images are belonging to different categories and these images are not specific domain. It is used in SIMPLIcity [14] and ALIPR [15] papers for research comparison. Hence we decided to use this image database to propose our algorithm and conduct experiments.

2.2 Feature Extraction

According to Liu and Wang and Zhang [13], image consists of redundant pixels which can be reduced in the pre-processing stage. In the pre-processing stage, the image is portioned into 2 × 2 blocks. The average color components of each image blocks are extracted as a feature vector in the RGB color space. This size is good for compromise between texture granularity, segmentation coarseness, computational complexity and it gives good result. The HSV color space is widely used for extracting color features due its ability to transform from RGB to HSV and vice versa. Since HSV color space has an advantage to produce a collection of colors that is also compact and complete, hence these features are represented as $\{f_1,f_2,f_3\}$.

To obtain the other three texture features from each color channel. We propose Haar wavelet transform to the L component; it is discontinuous and resembles a step function. It represents the energy in high frequency bands of the Haar wavelet transform [12]. After one-level wavelet transform, a 4 by 4 block is decomposed into four frequency bands, each band containing a 2 by 2 matrix of coefficients. Suppose the coefficients in the HL band are $\{c_k+i,\ c_k,\ l+1,\ l,\ c_k+1,\ l+1\}$. The feature of the block in the HL band is computed using equation1, these texture features are denoted as $\{t_1, t_2, t_3\}$.

$$f = \left[\frac{1}{4}\sum_{i=0}^{1}\sum_{j=0}^{1} c_{k+i,l+j}^{2}\right]^{\frac{1}{2}} \quad (1)$$

Finally we obtained 6-dimensional feature vectors; these features are used as input to Self- Organizing Maps for generating code words.

2.3 Self-Organizing Maps

The Self-Organizing Maps (SOM) algorithm is implemented to generate the code words that are spatially correlated to nearer to input vectors. The SOM are a type of unsupervised and iterative clustering algorithm of neural network for reduction of high-dimensional data space into lower dimensional data space for analysing the image content. Generally, SOM requires input vectors and the training algorithm returns representative code words that are spatially correlated to nearer to input vectors. Initially, these weights are randomly generated and have similar dimensions equivalent to the input vectors. In addition to input images, SOM requires random weights that have similar dimensions that input vectors. These random weights were generated and denoted by w_n, where n is the length of weight vector. The SOM algorithm is briefly discussed in the following steps [10], [11]:

- Each node weights w are initialized ($0 < w < 1$).
- An input vector is chosen at random from the set of training data and presented to the lattice.
- Traverse each node to find the BMU by calculating the most similar node weight to the input vector in feature space by using the Euclidean distance function. The Euclidean distance is given as:

$$\mathrm{Dist} = \sqrt{\sum_{i=0}^{n} \min_{1 \le j \le k}\left(\mathrm{I_i} - \mathrm{w_j}\right)^2} \quad (2)$$

where I the current input vector and w is the node's weight vector.

- Calculate the radius of the neighbourhood of the BMU. This value is set to the radius of the output map, but diminishes in each iteration
- Update the nodes in the neighborhood of BMU by calculating, which one is closer to the input vector as $\mathrm{c_{in}} = \mathrm{c_i(t)} + \theta(\mathrm{t})\alpha(\mathrm{t})\left(\mathrm{x_i(t)} - \mathrm{c_i(t)}\right)$. the closer node is to be the BMU, the more its weights get altered.

Here, t is the current iteration, $\mathrm{c_i(t)}$ and $\mathrm{x_i(t)}$ are the weight vector and target input vector at iteration t, whereas $\theta(\mathrm{t})$ and $\alpha(\mathrm{t})$are the neighbourhood function and learning rate due to the time.

- Repeat the above from step 2 for maximum iterations till weights vectors stabilize.

As soon as the weights vector stabilizes, the code word is generated. The code word size is determined based on number of experiments. These code words are used to represent feature vector discussed in the next section.

2.4 Feature Representation

The image feature sets are represented by n-dimensional feature space or feature vector. In this way, the domain dependent part of the whole image retrieval system is reduced to a minimum. The code words are the description of the original image obtained by SOM. We analysed and understood that the size of the code word is large to represent the original image. Therefore to reduce the dimensionality of the code word we used correlation coefficient to represent in the form of feature vector [17]. The advantage of correlation coefficient measures the strength and direction of a linear association between two variables. This evaluates the linear spatial relationships between color and texture primitives by using equation 3.

$$cr_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}} \quad (3)$$

where $(x_i - \bar{x})$ & $(y_i - \bar{y})$ is the difference of each code word minus the mean of code word. It gives the correlation coefficent as the mean of the products of the standard scores. The significance of new representaion is the measure of the stregth of linear dependence ranging from -1 to +1. A value of +1 is the result of a perfect positive relationship between two or more variables. Conversely, a value of -1 represents a perfect negative relatioship.

To be precise, let the code word c_f be represented by its image feature vectors of the form feature matrix F_m of the m^{th} feature of the input image with the positional differences is extracted and correlation co-efficient values of m^{th} features are represented into feature matrix as shown.

$$F_m = \begin{bmatrix} c_f(0,0) & c_f(0,1) & c_f(0,y-1) \\ c_f(1,0) & c_f(1,2) & c_f(1,y-1) \\ & . & \\ & . & \\ & . & \\ c_f(x-1,0) & c_f(x-1,0) & c_f(x-1,y-1) \end{bmatrix} \quad (4)$$

The values obtained with the feature matrix of each image are represeetned as the feature vector.

3. EXPERIMENTAL RESULTS

In this section, the results from the proposed method on low resolution Web-crawled miscellaneous image database [16] consisting of 10,000 images. The HSV and Haar Wavelet transforms are used for the color and texture feature extraction. The extracted features are given as input to the SOM algorithm to visualize and interpret large high-dimensional data sets. The image features are normalized to the range [-1, +1] to avoid the dominance of some features. In unsupervised learning, such as SOM, it is used for clustering the input data and find features inherent to the problem. The performance of SOM algorithm is studied by selecting different sized output layer network topology that is 4 x 4, 16 x 16, 64 x 64, 256 x 256 from top to bottom to generate codeword to represent an image semantically.

3.1 SOM Interpretation

The SOM encode and visualizes the HSV color space and Haar wavelet texture features in U-matrix representations. The landscape that represents the valleys or dark areas of the clusters of data and the mountains or light areas represents the boundaries between the clusters. A total of 718 features were extracted from the SOM analysis and these features are used to compute the correlation to obtained feature vector. These correlated features are stored and indexed to the original image in the database to find the distance between the query image and set of images. The rank of the image is measured based on the computational distance, the closest distance is displayed on the user interface in an increasing order.

We conducted the experiments to evaluate the precision of the proposed VBIRS system. The results have been recorded and first 20 closest images are shown in the figure 1. The recall is not measured in this study, because it is very difficult to calculate the recall of the system. It is a tedious job to browse the entire image database and specify the ground truth manually. Therefore we have considered top 20 output images to demonstrate the performance of the system. The recall is roughly estimated after scanning the top 100 images returned for some of the selected query image.

3.2 Similarity and Performance Measures

The set of images are represented by feature vectors of the form $I_j=\{f_{j1},f_{j2},\ldots,f_{jn}\}$ and a typical query by $I_q=\{f_{q1},f_{q2},\ldots,f_{qn}\}$ in the image database. The similarity measure is used to find the similar images from the image database. One of the measures of similarity is the cosine angle between the feature vector of the target image and the query image. The vectors of each vector of an image have observations and contain an entry for each distinct term in the entire image collection. The components in each vector are filled with weights computed for each term in the document collection. The cosine similarity measure is used to find the distance between query image and database images is as follows:

$$SC = \left(I_q, I_j\right) = \frac{\sum_{i=1}^{N} w_{iq} * w_{ij}}{\sqrt{\sum_{i=1}^{N}(w_{iq})^2} * \sqrt{\sum_{i=1}^{N}(w_{ij})^2}}$$

(5)

where, w_{iq} and w_{ij} are the weights of image features f_m in query image I_q and database image I_j respectively. The proposed cosine similarity measure quite simple and it takes less time to measure the distance in the large image database to retrieve the similar images.

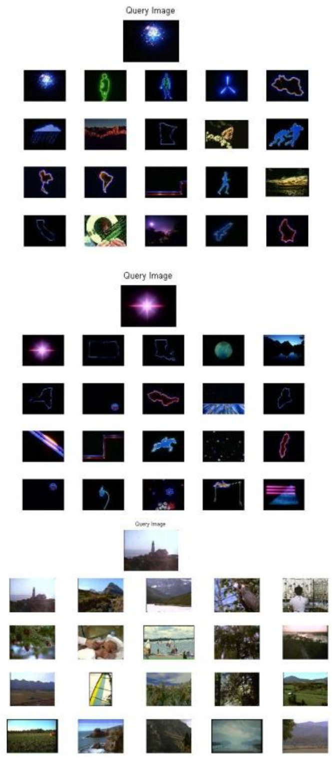
3.3 Results and Discussions

The proposed system has been used for feature extraction, image representation and similarity measures for test images. Features are extracted from the images and stored in the database. The correlated features correspond to semantic objects, allowing efficient indexing and retrieval.

The user starts a query by selecting the input image. The system returns a set of similar images; the majority of results appear to have high semantic relevance with the submitted image is shown in Fig 1.

The algorithm has been implemented on a Pentium(R) IV 3.00 GHz PC using Windows operating system. TO compute eh feature vectors for the 10,000 color images of size 128 x 85 and 128 x 96 in our general-purpose image database requires approximately 4 hours and 10 minutes. On average, 1.5 seconds is needed to compute the feature vector of an image. The matching speed is very fast. When the query image is in the database, it takes about 0.1 seconds of CPU time on average to sort all the images in the 10,000 image database using the cosine similarity measure. If the query image is not already in the database, 1.5 extra second of CPU time is spent to compute the feature vector from the query image.

Fig 1. Example of a query and set of retrieved relevant images



4. CONCLUSIONS

We proposed semantic based image representation to retrieve images from the large image database. The SOM is an iterative procedure used to generate code words for every image to represent cluster similar features efficiently. The results of our experiments show that the proposed technique is able to retrieve similar kind of image effectively from the large image database. To increase better retrieval performance is to do an extensive study of different feature representations to reduce quantization error and topological error rate to evaluate the complexity of the output space. As a vast collection of unclassified images are available on the internet are the major challenges of Web images. We conducted an Experiment on a WBIIS image database to demonstrate the efficiency and effectiveness of the proposed framework.

5. ACKNOWLEDGMENTS

We thank the Penn State, University Park, James Z. Wang Research Group for making the low resolution web-crawled image data available for this study.

6. REFERENCES

[1] Rui, Y., Huang, T.S., Chang, S.-F., 1999. Image retrieval: current techniques, promising directions, and open issues. Journal of Visual Communication and Image Representation 10 (1), 39-62.

[2] Flickner, M., Sawhney, H., Niblack, W., et al., 1995. Query by image and video content: the QBIC system. IEEE Computer September, 23-31.

[3] Pentland, A., Picard, R.W., Sclarof, S., 1994. Photobook: tools for content-based manipulation of image databases. In: Storage and Retrieval for Image and Video Databases II. In: SPIE Proceedings Series, Vol. 2185. San Jose, CA, USA.

[4] Minka, T.P., 1996. An image database browser that learns from user interaction. Master's thesis, M.I.T., Cambridge, MA.

[5] Michael J. Swain., Charles Frankel., and Vassilis Athitsos, “WebSeer: An Image Search Engine for the World Wide Web”, Technical Report 96-14, 1997.

[6] J. R. Smith, “Integrated Spatial and Feature Image Systems: Retrieval, Compression and Analysis”. PhD thesis, Graduate School of Arts and Sciences, Columbia University, February 1997.

[7] S. Sclaroff., L. Taycher., and M. La Cascia. “Imagerover: A content-based image browser for the world wide Web”. In Proceedings IEEE Workshop on Content-based Access of Image and Video Libraries, June ’97, 1997.

[8] Bach, J.R., Fuller, C., Gupta, A., et al., 1996. The Virage image search engine: an open framework for image management. In: Sethi, I.K., Jain, R.J. (Eds.), Storage and Retrieval for Image and Video Databases IV. In: SPIE Proceedings Series, Vol. 2670. San Jose, CA, USA.

[9] Koikkalainen, P., 1994. Progress with the tree-structured self organizing map. In: Cohn, A.G. (Ed.), 11th European Conference on Arti®cial Intelligence. European Committee for Arti®cial Intelligence (ECCAI). Wiley, New York.

[10] Koikkalainen, P., Oja, E., 1990. Self-organizing hierarchical feature maps. In: Proceedings of 1990 International Joint Conference on Neural Networks, Vol. II. IEEE, INNS, San Diego, CA.

[11] T.Kohonen, Self-Organizing Maps, Springer-verlag, New York, 1997

[12] Salton, G., McGill, M.J., 1983. Introduction to Modern Information Retrieval. In: Computer Science Series. Mc-Graw-Hill, New York.

[13] Li, J., Wang, J. Z. and Wiederhold, G., (2000), “Integrated Region Matching for Image Retrieval,” ACM Multimedia, p. 147-156.

[14] James Z. Wang, Jia Li and Gio Wiederhold, ``SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries,'' IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 9, pp. 947-963, 2001

[15] http://alipr.com/

[16] R. Datta, J. Li, and J. Z. Wang, “Algorithmic Inferencing of Aesthetics and Emotion in Natural Images: An Exposition”, Proc. IEEE ICIP, Special Session on Image Aesthetics, Mood and Emotion, San Diego, CA, 2008.

[17] J. L. Rodgers and W. A. Nicewander, “Thirteen ways to look at the correlation coefficient. The American Statistician”, 42(1):59–66, February 1988.