A NEURAL NETWORK APPROACH FOR LEARNING IMAGE SIMILARITY IN ADAPTIVE CBIR

P. Muneesawang
School. of Elect. and Info. Engin.
University of Sydney
Sydney, Australia

L. Guan
Dept. of Elect. and Comp. Engin
Ryerson Polytechnic University
Toronto, Canada

Abstract - In this paper the adoption of neural network techniques is studied for the purpose of image retrieval. More specifically, we propose an adaptive retrieval system which incorporates learning capability into the image retrieval module where the network weights represent the adaptivity. This system can learn users' notions of similarity between images through the continual relevance feedback from the users. Accordingly it makes the proper adjustment to improve performance. This retrieval system has demonstrated its effectiveness in performance. It is confirmed by simulations conducted for applications such as texture retrieval and retrieval of the DCT compressed images.

INTRODUCTION

In this paper the adoption of a machine-learning approach is proposed for *image matching* in content-based image retrieval (CBIR). This learning approach has one main advantage over the traditional retrieval approaches: it allows the retrieval system to solve the problem of fuzzy understanding of users' goals that are not realized by a one-shot retrieval procedure. Specifically, we propose the adoption of neural network techniques [1][2] for this task in view of their learning capability and their ability to simulate universal mapping. These techniques allow the users to directly modify the query characteristics by specifying their desired image attributes in the form of training examples. Within this framework, the retrieval system can adjust its strategy to progressively model the notion of image similarity through continual relevance feedback from the user. In fact it uses individual user choices to decide relevance.

We have adopted a Self-Organizing Tree Map (SOTM) [1] and a Learning Vector Quantization (LVQ) [3] to generate prototypes in the form of local data clustering. This is used for approximating the distribution of the relevant samples which are associated with the training data. The resulting prototypes are then incorporated into a *single-pass* radial basis function network (RBFN) for ranking of an image database. In the experiments, the proposed interactive system has been applied to a compressed domain image retrieval as well as to the retrieval system that use image features extrated from uncompressed images. The comparison with other methods is provided.

0-7803-7025-2/01/$10.00 ©2001 IEEE.

GENERATION OF PROTOTYPES USING SOTM/LVQ ALGORITHMS

Traditional query refinement strategy [4] is implemented as an iterative search strategy which uses the information provided by the user to update the initial submited query. However, this produces a single query which only allows for global similarity evaluation, but does not adapt well to the local context defined by the current query. Here, we implement the query refinement strategy based on *local clustering* which aims at optimizing the current search. We propose the adoption of an efficient data clustering technique, the SOTM [1] algorithm to achieve this purpose. A set of *relevant* images is treated as training data, to create the weight vectors called 'prototypes' which are found locally in the input space. This follows the distribution of input data associated with the relevant images. We then adopt LVQ to modify the prototypes by negative samples (i.e., non-relevant images) to improve the clustering performance.

(1). Unsupervised Growing of a New Cluster. The SOTM [1] provides construction of unsupervised suitable feature clustering. It is more effective than the K-mean and the self organizing feature map (SOM) algorithms particulary when the input space has a high dimensionality (i.e., sparse data). Thus, SOTM is chosen in our work to locate cluster centers in the high dimensional space of image features.

To carry out the prototypes using SOTM/LVQ algorithms, we are given a set of training samples corresponding to retrieved images from a previous search operation. We formally denote this training data with two sets of vectors: positive sample set (relevant images) $\mathbf{X}^+ = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathcal{R}^P$, and negative sample set (non-relevant images) $\mathbf{X}^- = \{\mathbf{x}'_1, \ldots, \mathbf{x}'_M\} \subset \mathcal{R}^P$. We then assign each input vector in $\mathbf{X}^+$ into an SOTM algorithm to create the weight vectors, whereas the vectors in $\mathbf{X}^-$ are used in the LVQ algorithm for further adjustment of the weight vectors.

SOTM algorithm is summarized as follows [1]:

(i). *Initialization.* Choose the root nodes $\{\mathbf{w}_i\}_{j=1}^{l}$ from the available set of input vectors $\{\mathbf{x}_i\}_{i=1}^{N}$ in a random manner.

(ii). *Similarity matching.* Randomly select a new data point $\mathbf{x}$, and find the best-matching (winning) neuron at time step t by using the minimum-distance Euclidean criterion:

$$\mathbf{w}_{j^*} = \arg\min_j ||\mathbf{x}(t) - \mathbf{w}_j||, \quad j = 1, 2, \ldots, l \tag{1}$$

(iii). *Updating.* **If** $||\mathbf{x}(t) - \mathbf{w}_{j^*}|| \leq H(t)$, where $H(t)$ is the hierarchy function used to control the levels of the tree, **then** assign $\mathbf{x}(t)$ to the j-th cluster, and adjust the synaptic weight vector according to the reinforced learning rule:

$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \alpha(t)[\mathbf{x}(t) - \mathbf{w}_j(t)], \tag{2}$$

where $\alpha(t)$ is the learning rate, which decreases monotonically with time, $0 < \alpha(t) < 1$. **Else** form a new subnode starting with $\mathbf{x}$.

(iv). *Continuation.* Continue with step (ii) until no noticeable changes in the feature map are observed.

The SOTM algorithm obtains a new set of cluster centers $V_0 = \{\mathbf{v}_1, \ldots, \mathbf{v}_k, \ldots, \mathbf{v}_K\}$, where the number of centers K is controlled by the function $H(t)$. In the experiment, $H(t)$ was initialized by the norm of the training vectors in X^+, and was reduced linearly.

(2). Cluster Modification. At this stage, the negative samples in $\mathbf{X}^-$ are used to tune the decision boundaries of the initial set of prototypes V_0. We employ the antireinforced learning rule in the LVQ algorithm [3] to perform this operation. The algorithm starts with randomly choosing input vector $\mathbf{x}'_m(t)$ from the training set $\{\mathbf{x}'_m\}_{m=1}^M$. Then, classify $\mathbf{x}'_m(t)$ to node $\mathbf{v}_i$ if $||\mathbf{x}'_m(t) - \mathbf{v}_i|| < ||\mathbf{x}'_m(t) - \mathbf{v}_k||, \forall k \neq i$, and apply the antireinforced learning rule to the corresponding cluster centers as follows:

$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) - \eta(t)[\mathbf{x}'_m(t) - \mathbf{v}_i(t)], \tag{3}$$

where $\eta(t)$ is the learning constant which decreases monotonically with the number of iterations t, $0 \leq \eta(t) \leq 1$. After several passes through the training data, the node vectors typically converge, and the modification process is complete. Note that we desire to move the prototypes slightly.

RANKING IMAGES USING RADIAL BASIS FUNCTION NETWORK (RBFN)

Having the prototypes to describe the local clusters of relevant images, the next step is to use them in a new search for retrieval. To obtain a new set of retrieved images, a process of evaluating similarity between each prototype and the input vectors corresponding to images in the database is performed . We proposed a single-pass RBF network to perform this operation using its nonlinear discrimination capability. Unlike traditional RBF network models that require iterative learning, the proposed network requires only a single pass of processing to allow rapid evaluation of image similarity in an interactive session.

To carry out function approximation using RBFN, we are given a set of prototypes $V_0 = \{\mathbf{v}_1, \ldots, \mathbf{v}_k, \ldots, \mathbf{v}_K\} \subset \mathcal{R}^P$ from the SOTM/VQ algorithms. We then assign each input vector $\mathbf{v}_k$ as the center of the corresponding Gaussian kernel in the network. In other words, the RBFN uses p-D Gaussian distributions to describe the shapes of data clusters indicating the relevant class. For an arbitrary input vector $\mathbf{x}$, the output of the k-th RBF unit is given by

$$G_k(\mathbf{x}, \mathbf{v}_k, \sigma_k) = \exp\left[-\frac{(\mathbf{x} - \mathbf{v}_k)^T(\mathbf{x} - \mathbf{v}_k)}{2\sigma_k^2}\right] \tag{4}$$

where σ_k is a smoothing parameter defined as,

$$\sigma_k = \beta \cdot \min_i ||\mathbf{v}_k - \mathbf{v}_i||, \quad i = 1, 2, \ldots, K \tag{5}$$

and $\beta = 0.5$ being an overlapping factor. The estimated function output $F(\mathbf{x})$ for $\mathbf{x}$ is then given as:

$$F(\mathbf{x}) = \sum_{k=1}^{K} G_k(\mathbf{x}, \mathbf{v}_k, \sigma_k) \tag{6}$$

Intuitively, the RBFN performs similarity evaluation by linearly combining the output G_k. If the current input vector $\mathbf{x}$ (associated with an image in the database) is close to one of the RBF centers $\mathbf{v}_k$ (i.e., the k-th prototype) in a Euclidean sense, the corresponding RBF unit G_k given by Eq (4) will increase, and the estimated output $F(\mathbf{x})$ will also increase indicating the greater similarity of the corresponding image. On the other hand, those input vectors that are far away from each of $\mathbf{v}_k, k = 1, \ldots, K$ do not appreciably contribute to the summation due to the exponentially decaying weighting function.

EXPERIMENTS AND APPLICATIONS

Texture Retrieval on the Brodatz Database

This method was compared with the traditional relevance feedback method (RFM) [4], using the Brodatz texture database [5]. The database containes 1856 images which were obtained from 116 different texture classes, where each class contains 16 images. Each texture image in this database is described by a 48-dimensional feature vector that characterizes the coefficients after applying the Gabor wavelet (GW) transform to the image [6].

In the simulation studies, a total of 116 images, one from each class, was selected as the query images. For each query, the top 16 images were retrieved to evaluate retrieval performance. The performance was measured in terms of the retrieval rate [6] defined as the average percentage number of images belonging to the same class as the query in the top 16 matches.

The proposed neural network (NN) method and the relevance feedback method (RFM) have been tried. The retrieval performance is summarized in Table 1, showing very satisfactory convergence speeds and retrieval accuracies (where relevance feedback was provided upto 3 iterations). It can be seen that all learning methods demonstrated significant improvement across the tasks. The retrieval rate was improved from 73.7% (normalized Euclidean measure) to 87.6% with the application of the proposed NN method, i.e., more than 14 correct images were presented out of 16. This result shows that the NN method performs substantially better than RFM which provides a retreival performance of 78.5%.

In Table 1 the performance of the NN method was based on the SOTM/VQ algorithm with the muximum number of prototypes equal to 8 (this is controlled by the hierarchy function $H(t)$). After varying the number of the prototypes from 2 to 15, we observed that the retrieval performances were gradually changed in a range of 86.3% to 87.6%.

Method	0 Iter.	1 Iter.	2 Iter.	3 Iter.
NN	73.71	83.72	86.53	87.55
RFM	67.10	75.74	77.76	78.46

Table 1: Average retrieval rate (%) of 116 query images on the Brodatz database.

Application to Compressed Domain Image Retrieval

In this experiment we applied the proposed interactive method to a compressed domain image retrieval system. Specifically, the matching process is directly performed on the DCT domain to avoid the costly operation of decompression. The image database is a single uncategorized database that consists of nearly 4700 JPEG photographs [7] covering a broad range of categories.

The image feature used here is based on an energy histogram of a DCT coefficient originally proposed in [8]. An energy histogram of DCT coefficients is obtained by counting the number of times a particular coefficient value appears in the 8×8 block. For this experiment, the energy histogram features are based on the four lower frequency DCT coefficients in the upper left corner of the DCT block. Separate energy histograms are constructed for the DC and AC coefficients of each of the color channels, and 30 bins are used for each histogram.

A total of 10 query images (each was different to the other) were manually selected to ensure that the similar images by human vision were properly identified in the database. Since the number of relevant images for a particular query could not be easily identified from the database, the performance was measured by counting the number of relevant images that hit the desired target in the top 15 matches. Table 2 details the retrieval results. It can be seen that the number of relevant images increased considerably by using the learning approach.

Typical retrieval sessions are shown in Figure 1, where Figures 1(a) and 1(b) show the 15 top-matched images before and after the application of the learning technique respectively. The improvement given by the proposed method is apparent.

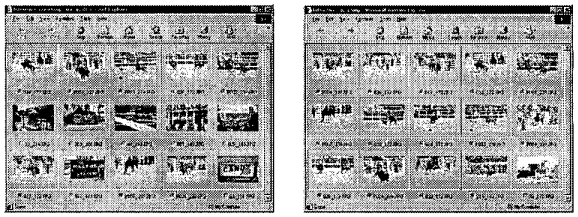

Figure 1: left: (a) the retrieval results before learning similarity in answer to the query 024_172.JPG; right: (b) the retrieval resuls after learningsimilarity.

Query	0 Iter.	1 Iter.	2 Iter.	Query	0 Iters	1 Iter.	2 Iter.
024_172.JPEG	8	14	14	046_160.JPEG	4	6	6
001_181.JPEG	5	10	10	065_200.JPEG	3	7	7
P39_226.JPEG	3	5	6	058_143.JPEG	8	11	10
P73_236.JPEG	2	5	7	076_126.JPEG	4	6	9
32_178.JPEG	6	11	12	040_123.JPEG	4	10	9

Table 2: Retrieval results on the JPEG database (4678 images) using the DCT energy histogram feature. The table shows the number of relevant image in the top 15 matches.

CONCLUSION

We have proposed the adoption of neural network techniques for characterizing the behavior of human users in an interactive session where relevance feedback is applied. With the SOTM/VQ algorithms, a more effective local analysis of relevance is provided than with another better-known model: query modification approach. The network structure also includes a single-pass RBFN to perform a more accurate non-linear evaluation of image similarity for ranking purposes. Based on a simulation performance comparison, this proposed learning-based approach appears to be highly effective for retrieval application on compressed and uncompressed-domain image retrievals.

References

[1] H.Kong and L.Guan, "Self-organizing tree map for eliminating impulse noise with random intensity distributions", *J. of Electronic Imaging*, vol.7/1, pp.36-44, 1998.

[2] T. Sigitani, Y. Liguni, H. Maeda, "Image interpolation for progressive transmission by using radial basis function networks", *IEEE Trans. on Neural Networks*, vol.10/2, pp. 381-390, 1999.

[3] S. Haykin, *Neural Networks: a Comprehensive Foundation*, Prentice Hall, NJ, 1999.

[4] Y. Rui, T.S. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS", *Proc. IEEE Int. Conf. on Image Processing*, pp. 815-818, 1997.

[5] http://vivaldi.ece.ucsb.edu/users/wei/codes/thml, 1999.

[6] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data", *IEEE Trans. of Pattern Analysis and Machine Intelligence*, vol. 18/8, pp. 837-842, 1996.

[7] Media Graphics International, Photo Gallery 5,000 Vol.1 CDROM, http://www.media-graphics.net.

[8] J.A. Lay and L. Guan, "Image retrieval based on energy histogram of the low frequency DCT coefficients", *Proc. of the ICASSP*, pp.3009-3012, 1999.