

Relevance Feedback Using RBF for Content Based Image Retrieval

Guizhi Li, Changsheng Zhou , Wei Wang and Yahui Liu

Computer Center, Beijing Information and Science and Technology University, China

lgz403@sina.com

Keywords: RBF; Relevance Feedback; Image Retrieval

Abstract. Conventional approaches to image retrieval are based on the assumption that relevant images are physically near the query image in some feature space. However, semantically related images are often scattered across several visual clusters. This leads to adapting multiple queries to represent a query in the feature space. Therefore, it is necessary to handle disjunctive queries in the feature space. In this paper, a new content-based image retrieval method with relevance feedback technique using RBF neural network learning is proposed. The method transfers the process of relevance feedback into a learning problem of RBF neural network. RBFNN can describe the distribution of positive feedback sample images in feature space with a set of neighboring clusters produced through constructing neural network, for accurately reflecting their semantic relevance. The performance of the method using RBFNN is evaluated on a database of 10,000 images. Experimental results demonstrate the effectiveness of the proposed method.

Introduction

Content-based image retrieval (CBIR) has become one of the most active research areas in multimedia signal processing. It is well known that the performance of content-based image retrieval (CBIR) systems is mainly limited by the semantic gap between low-level features and high-level concepts. Retrieving images from a large database according to the semantic meanings is a challenging problem, due to the lack of mature artificial intelligence mapping semantic meanings to low-level descriptors.

Relevance feedback techniques have been widely used in CBIR to bridge the semantic gap between low-level features and high-level semantics[1]. Relevance feedback (RF) is an interactive technique in the procedure of CBIR, which is originally used in the traditional text-based information retrieval systems. A relevance feedback-based approach allows the user to label the returned images as relevant or irrelevant. Such labeled examples are further used to refine retrieval results by short-term learning or long-term learning techniques.

Many relevance feedback algorithms such as query refinement[2], re-weighting[1], Bayesian or probabilistic learning[3], optimal learning over heuristic-based feature weighting[4], artificial neural networks[5], and kernel-based learning[6], etc., have been adopted in CBIR systems and demonstrated considerable performance improvement. Current approaches to CBIR assume that relevant images are physically near the query image in some feature space regardless of visual features. However, the similarity between images perceived by humans does not necessarily correlate with the distance between them in the feature space. That is, semantically relevant images might be spread out in the feature space and be scattered in several clusters rather than one. In this case, traditional relevant feedback approaches do not work well when shifting the query center by linear combination of the relevant images.

In this paper, we propose a relevance feedback technique based on NBFNN for content-based images, which transfers the process of relevance feedback into a neural network learning problem. By constructing neural network that is constituted of the neighboring clusters of sample images, the distribution of user feedback sample images is described in feature space, so as to reflect the relevance of sample images in semantic contents. In particular, constructing neural network is dynamic, that is, neural network constructed depends on which images are retrieved in response to the query.

The rest of this paper is organized as follows. In section II we describe the proposed RBF network and its discriminant function. The principal of the self-growing RBF neural network is introduced in section III. The proposed relevance feedback based on RBF is presented in section IV. Our experimental results are given in Section V. Finally, Section VI gives some concluding remarks of this study.

RBF Neural Network

A RBF network usually consists of three layers: the input layer, the RBF layer (hidden layer) and the output layer. The input layer simply transfers the input vector $x = [x_1, x_2, \dots, x_n]^T$ through scalar weights to the next layer.

In most applications, the scalar weights are assigned unity values. Thus the whole input vector appears to each neuron in the hidden layer. Each hidden nodes perform the radial basis function over the incoming vector that appears at the input of each RBF neuron. The output layer yields a vector for k outputs by linear combination of the outputs of the hidden nodes to produce the final output. Fig.1 presents the structure of a single output RBF network. The network output can be obtained by:

$$y = f(x) = \sum_{i=1}^k w_i \phi_i(x) . \quad (1)$$

where $x = [x_1, x_2, \dots, x_n]^T$ is the input vector, $f(x)$ is the final output, $\phi_i(\cdot)$ denotes the radial basis function of the i th hidden node, w_i presents the hidden-to-output weight corresponding to the i th hidden node, and k is the total number of hidden nodes.

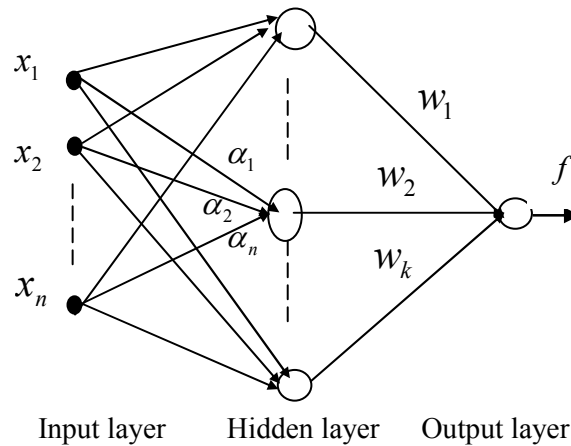


Fig. 1 The RBF neural network architecture

A radial basis function is a multidimensional function that describes the distance between a given input vector and a pre-defined center vector. There are different types of radial basis function. A normalized Gaussian function usually used as the radial function, that is:

$$\phi_i(x) = \exp \left(- \sum_{j=1}^n \alpha_j (x_j - u_{ij})^2 / (2\sigma_i^2) \right) . \quad (2)$$

Where $u_i \in R^n$ denotes the center of the Gaussian function and σ_i denotes its width. The α_j denotes the relative importance of different feature components.

The Self-Growing RBF Neural Network

Since the number and location of centers in the hidden layer will influence the performance of the RBF network directly, it is very crucial to determine the parameters of Gaussian center. To solve this problem, the self-growing RBF neural network techniques have been developed. These approaches start with one or a few RBFs and generate new basis functions by recursively applying the input training data.

The rationale of this learning is described as follows: the hidden layer starts with no hidden node. The first RBF node center u_1 is set by choosing one data randomly from input data sample. The value of Euclidean 2-norm distance between u_1 and the next input sample x , is compared with ε . If it is greater, a new cluster which is centered at the location defined by x is created as u_2 , otherwise the elements of u_1 are updated as:

$$u_1 = u_1 + \eta \|x - u_1\| \quad (3)$$

Where $\|\cdot\|$ denotes the Euclidean distance, $0 < \eta < 1$ is the updating ratio. The criterion ensures that a new neuron is only added if input data is sufficiently far from the existing neurons. Thus, this procedure carried out to the remaining training sample. Consequently, the number of clusters grows, and RBF nodes center self-adjust continuously until all of the samples are processed.

Relevance Feedback Based on RBF

In the current approach, the query is modeled according to the local clustering distributed by the relevant samples. Instead of using a single modified query, the new alternative uses a series of prototypes as the models of relevant images. We propose learning algorithms which enable the RBF network to progressively model the notion of image similarity for effective searching. The image matching process is initiated when the user supplies a query image and the system retrieves the images in the databases which are closest to the query image. From these images the user selects those as relevant which are most similar to the current query image, while the rest are regarded as non-relevant. The feature vectors these images are incorporated as training data for the RBF network to modify the centers and widths. The re-estimated RBF model is then used to evaluate the perceptual similarity in a new search, and the above process is repeated until the user is satisfied with the retrieval results.

The process of RBF neural network learning is divided into two stages. The first stage is to construct self-growing BRF neural network using relevance images, every neuron of which is named neighboring cluster. The second stage is to use the negative samples to modify the model.

Network Growth Technique. Our objective is to split a single query into multiple clusters. Instead of modifying a single query we use a set of models to characterize the query. We utilize positive samples, to train the self-growing BRF neural network since these samples represent the desired images. The modification of the cluster centers is usually carried out by an iterative process. The algorithm can be summarized as follows:

(1) Assuming that $\{u_i\}_{i=1}^k$ denote the set of k cluster centers, which are generated from previous iterations. Taking a new input positive sample $x' \in R^n$, where n corresponds the dimension of input spaces, to calculate the distances with each cluster $\|x' - u_i\|$ for $i = 1, \dots, k$.

(2) Then the best-matching cluster center is identified by the condition:

$$\|x - u_j\| \leq \|x - u_i\|, \forall i \neq j \quad (4)$$

(3) Comparing $\|x - u_j\|$ with the distance criterion parameter ε . If it is greater than threshold, then a new cluster center u_{k+1} is created at the position of the sample point x . Otherwise it means this input vector belongs to the j th cluster and the cluster center u_j is updated by:

$$u_j = u_j + \eta \|x - u_j\| \quad (5)$$

(4) Repeating the whole procedure while taking the next input positive sample until all of the relevant samples are processed.

RBF Learning. Then we use the negative samples $x'' \in R^n$ to modify the model because they are the closest vectors to the query and the positive samples according to the distance calculation of the previous search. To achieve this, we adopt the anti-reinforced learning rule of learning vector quantization algorithm to move each u_i slightly away from negative samples, so as to improve the quality of the classification regions. To be precise, let x'' be the closest point to the model u_i . The model is modified according to the following learning rule:

$$u_i = u_i - \eta_2 (x'' - u_i) \quad (6)$$

Where η_2 is the learning constant which decreases with the number of iterations. Note that it is desirable for the process to move the models slightly away from negative samples.

The similarity evaluation is then performed based on linear combination of all the responses from each Gaussian RBF units. The overall output $f(x)$ for an input data x from the database functions as a similarity measure, namely, those images x in the database with higher values are more likely to be relevant to the user.

Selection of RBF Parameters. Not all feature variables are equally important in similarity judgment. Thus it is important to indicate the relative importance of various feature variables by weighting them. Weight schemes need to be estimated based on judgments of relevance, and the nature of the query. Where α_j in Eq.2 is a relevance weight which is derived from the variance of the positive samples as:

$$\alpha_j = \begin{cases} 1 & \xi_p = 0 \\ 1/\xi_p & \xi_p > 0 \end{cases} \quad (7)$$

Where ξ_p is the standard deviation of the positive sample sequence. This weighting criterion is based on the assumption that if a particular feature is important, then all relevant samples should have a very similar value for this feature, i.e., the sample variance in the relevant set is small.

Different positive images may have different degrees of relevance, thus contributing unequally to the user information need. At the feedback loop stage, the user specifies a score value v_i for each relevant image x_i . Later, after the RBF training stage of the current iteration, if x has become the k th point of u_j , then the weight of u_j is:

$$m_j = \sum_{i=1}^k v_i \quad (8)$$

Then w_j is the normalized weight of the k th cluster:

$$w_j = m_j / \sum_{i=1}^k m_i \quad (9)$$

The determination of the RBF unit width is given by:

$$\sigma_i = \delta \cdot \min_j \|u_i - u_j\| \quad j = 1, 2, \dots, k \quad (10)$$

Where δ is a factor that controls the overlapping of different RBF units.

Experiment Results

The image database used in the experiment contains 10,000 color images of 100 different categories obtained from the Corel Gallery product. In our system, we use two types of visual features: color and texture. Color histogram, color moments, and color auto-correlogram are used as the representation for color features. Gabor wavelet and wavelet moments are used as the texture features representation.

In the experiments, we select 100 query images, one from each category, and evaluate the retrieval quality for a sequence of iterations starting with these initial queries. For each query, the top 20 images were retrieved to evaluate the retrieval performance. We perform 5 feedback iterations in addition to the initial query. All the measurements are averaged over 100 queries. Precision and recall curve is used to measure the retrieval performance. Precision is defined as the number of retrieved relevant images over the number of total retrieved images. Recall is defined as the number of retrieved relevant images over the total number of relevant images.

Fig.2 illustrates the precision and recall graphs over the different iterations. From the graph, we can observe that the precision and the recall increase at each iteration. The precision and recall are 0.5 and 0.1 at first iteration and 0.8 and 0.16 at fifth iteration, respectively. The retrieval quality increases quickly in the initial stage. At the following iterations there are minor increases in the retrieval quality. This ensures that our method converges quickly to the user's true information need.

Fig.3 illustrates the precision and recall graphs over the different number of retrieved images. According to Fig.3, the precision decreases and the recall increases with increase of the number of retrieved images. This is reasonable, because with the increase of the number of retrieved images, more and more images that are irrelevant to the query are involved in the retrieved images, which have negative impact on the precision. When the value is 100, since the number of retrieved images is equal to the number of relevant images, the value of precision and recall are the same.

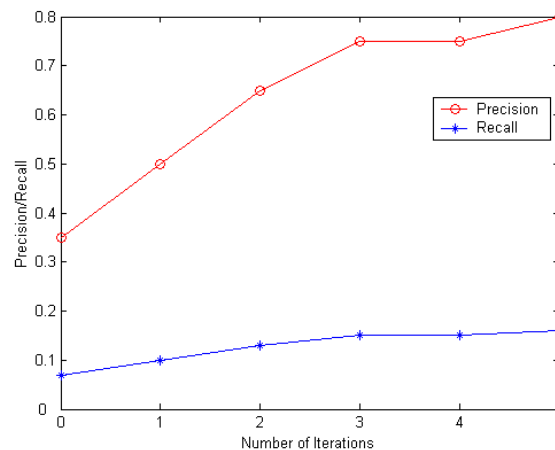


Fig.2 Precision and recall graphs over the number of iterations

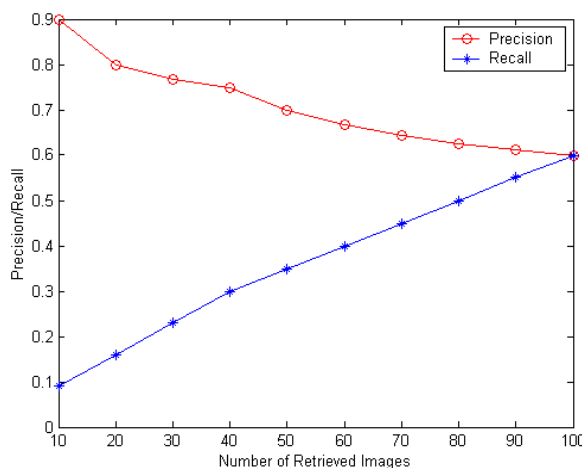


Fig.3 Precision and recall graph over the number of retrieved images

Conclusion

This paper presents a new content-based image retrieval method with relevance feedback technique using RBF neural network learning. The method transfers the process of relevance feedback into a learning problem of RBF neural network. RBFNN can describe the distribution of positive feedback sample images in feature space with a set of neighboring clusters produced During the feedback iterations, for accurately reflecting their semantic relevance. Experimental results demonstrate that it is an effective method to improve semantic image retrieval performance.

Acknowledgment

This work was supported by a grant from General program of science and technology development project of Beijing Municipal Education Commission. (No.51072013)

References

- [1] Y. Rui T. S. Huang, M. Ortega, S. Mehrotra, Relevance feedback: a power tool for interactive content-base image tetrieval, IEEE Transactions on Circuits and Video Technology, vol 8, no.5, pp.644-655, 1998.
- [2] K. Chandramouli, T. Kliegr, J. Nemrava, V. Svatek, and E. Izquierdo, Query refinement and user relevance feedback for contextualized image retrieval, In Proceeding of Visual Information Engineering, pp. 453-458, 2008.
- [3] M.M.Rahman, P.Bhattacharya, B.C.Desai, A framework for medical image retrieval using machine learning and statistical similarity matching techniques with relevance feedback, IEEE Transactions on Information Technology in Biomedicine, vol 11, No.1, pp.58-69, 2007.
- [4] Y. Ishikawa, R. Subramanya, MindReader: query database through multiple examples, Proc. Of Int. Conf. on Very Large Data Bases, New York, USA,1998.
- [5] P. Muneesawang and L. Guan, Automatic machine interactions for content-based image retrieval using a self-organizing tree map architecture, IEEE Trans. on Neural Networks, vol. 13, no. 4, pp. 821-834,July 2002..
- [6] R.J.Liu, Y.H.Wang, T. Baba, D.Masumoto, S. Nagata, SVM-based active feedback in image retrieval using clustering and unlabeled data, Pattern Recognition, vol.41,pp.2645-2655,2008.