IMAGE RETRIEVAL WITH EMBEDDED SUB-CLASS INFORMATION USING GAUSSIAN MIXTURE MODELS

P. Muneesawang †, and L. Guan ‡ 

† Dept. of Electrical and Computer Engineering, Naresuan University, Thailand. 
‡ Dept. of Electrical and Computer Engineering, Ryerson University, Toronto, Canada.

ABSTRACT
This paper describes content-based image retrieval techniques within the relevance feedback framework. The Gaussian mixture model (GMM) is used to characterize sub-class information to increase retrieval accuracy and reduce number of interactions during a query session. The implementation of GMM is based on the radial basis function using a new learning algorithm that can cope with small training samples in the relevance feedback cycle. The proposed retrieval system is successfully applied to image databases of very large sizes, and experimental results show that the proposed system competes favorably with the other recently proposed interactive systems.

1. INTRODUCTION

In image retrieval, particularly in general image collection, the relevancy of images to a specific query is appropriately characterized by a multiple-class modeling approach. For example, when a user has a query for a “PLANE”, she or he may wish to have any image containing planes, as shown in Fig. 1. The semantic of “PLANE” is clearly described by a variety of models, which are correlated, but each of which has its own local characteristics. The difficulty in characterizing image relevancy, then, is identifying the local context associated with each of the “sub-classes” within the class “PLANE”. Human beings utilize multiple types of modeling information to acquire and develop their understanding about image similarity. To obtain more accurate, robust, and natural characterizations, a computer must generate a reasonable definition of what humans regard as significant features. Through Human-Computer Interaction (HCI), computers do acquire knowledge of novel features which are significant but have not been explicitly specified in the training data. This implicit information constitutes subclasses within the query, permitting better generalization. In this paper, a mixture of Gaussian models is used, via the radial basis function (RBF) network, to represent multiple types of model information for the recognition and presentation of images by human beings and computers.

Section 2 of this paper will review the HCI techniques that are based on the relevance feedback for image retrieval. However, most of the previous attempts have only concerned with the single-class paradigm, in which a query images is described by one model, which is then associated with only a particular location in the input space. Furthermore, the similarity function is based on a single matrix. The combination gives rise to a classifier for a single cluster, which cannot fully exploit the local data information. We will propose in Section 3 a mixture of Gaussian models for interactive retrieval that enables the learning systems to take advantage of the information from multiple sub-classes. The proposed learning system utilizes a highly local characterization of image relevancy in the form of superposition of different local models as the classifier. As a result, local data distributions can be sufficiently exploited to achieve rapid performance improvements. Compared to the previous attempts, the current system specializes in effective learning using a very small set of input samples. Because of its quick-learning capability, significant improvement in retrieval precision can be immediately achieved within one to two feedback iterations. This is confirmed by the experimental results in Section 4 with the application to the Corel image database.

Fig. 1. Example images representing semantic of “PLANE”.

2. SINGLE CLASS LEARNING METHODS

The idea of deriving semantic in accordance with user’s preference is to create a learning-based system. In general, learning systems implement a mapping \( f_s : \mathbb{R}^n \rightarrow \mathbb{R} \) which is given by:

\[
y_s = f(x)\tag{1}
\]

where \( x = [x_1, ..., x_p]^T \in \mathbb{R}^p \) is the input vector corresponding to an image in the database. The main procedure is to obtain the mapping function \( f(x) \) from a small set of training samples \( T = \{ (x_1, l_1), (x_2, l_2), ..., (x_n, l_n) \} \), where the class label \( l_i \) can be in binary or non-binary forms which are defined by the user feedback.

As in the most content-based image (CBIR) systems (e.g., MARS [1]), the mapping function is obtained by a weight distance:

\[
f(x, x_q) = (x - x_q)^T W (x - x_q) \tag{2}
\]

where \( x_q \) is the feature vector of the query image, and \( W \) specifies the weight parameters of the function by its block-diagonal matrix form,

\[
W = \text{diag}[w_1, w_2, ..., w_p]. \tag{3}
\]
The weight parameters $w_i, i = 1, \ldots, p$ are estimated by the standard deviation criterion from the training set $T$. Further, this model can be enhanced by the optimum solutions for the query vector $x_q$ and the weight matrix $W$ as demonstrated in [2].

While the above models serve as the principle design for the learning-based CBIR systems, they offer very limited discrimination power, as the systems restrict themselves to a quadratic form, which cannot cope with a complex decision boundary. To overcome the problem, Munneawang and Guan [3] proposed a more complex algorithm employing a nonlinear kernel of the one-dimensional radial basis function (RBF). Compared to the conventional quadratic function, and the limited adaptivity allowed by its weighted form, the RBF approach offers an expanded set of adjustable parameters, in the form of RBF centers and widths:

$$f(x, z) = \sum_{i=1}^{p} \exp\left(\frac{(x_i - z_i)^2}{2\sigma_i^2}\right)$$

where $(z_i, \sigma_i)_{i=1}^{p}$ are the RBF centers and widths.

The methods outlined above are the basic for the learning-based CBIR systems based on a single model. This constructs $f$ by a supervised learning for the feature weighting scheme and/or a reformulation of a query model. In the classification context, we view this model as a single cluster in the feature space. For multiple models, however, it may be an advantage to partition the input space into multiple subspaces—an inherent strategy of local modeling techniques. We therefore adopt a Local Model Network (LMN) [4] to achieve this purpose. This is described in the following section.

3. LEARNING WITH GAUSSIAN MIXTURE MODEL

We adopt the LMN demonstrated in [4] to approximate the model function $f(x)$. Instead of using a single model function, the current work estimated the mapping function $f(x)$ by the superposition of different local models:

$$f(x) = \sum_{i=1}^{n_M} \lambda_i f_i(x, z_i) \equiv \sum_{i=1}^{n_M} \lambda_i \phi_i(||x - z_i||)$$

where $f_i(x, z_i)$ is the $i$-th local model function, and $\lambda_i$ is its associated linear weight. The advantage of this network used in the current application is that it finds the input-to-output mapping using local approximators; consequently, the underlying basis function responds only to a small region of the input space where the function is centered, e.g., a Gaussian response, $\phi_i(d) = \exp(-d^2/2)$ where:

$$d(x, z_i, \sigma_i) = \sqrt{(x - z_i)^T \sigma_i^{-2} (x - z_i)}$$

This allows local evaluation for image similarity matching.

Typically, the parameters to learn for the LMN are the set of linear weight $\lambda_i$, the center $z_i$, and the width $\sigma_i$ for each approximator, $\phi_i, i \in \{1, \ldots, n_M\}$. However, the theoretical investigations and practical results indicate that the choice of center $z_i$ is most significant in the performance of the LMN [4]. Learning methods such as the least mean square (LMS) with randomly selected centers [5] and the orthogonal LMS [6] are commonly used to solve this problem, where sufficient training samples are provided. In the interactive retrieval application, however, only small set of training samples are available. In addition, training images are usually highly correlated, which may be ‘ill-conditioned’, owing to the near-linear dependency caused by the fact “some centers are too close to one another”.

In view of these problems, we proposed a new learning algorithm for the LMN, and referred to as adaptive radial basis function network (ARBFN). Our proposed learning strategy for the ARBFN is based on the following points:

- Formulate and solve the local approximator $\phi(\cdot)$ from available positive samples.
- In order to take advantage of negative samples to improved decision boundary, we obtain shifting centers, instead of employing linear weights.
- In order to obtain a dynamic weighting scheme, we replace the Euclidean norm in $\phi(||x - z_i||)$, with the weighted Euclidean, which provides a new form of local approximator, $\phi(||x - z_i||, \lambda_i)$, where adjustable metric parameters are estimated according to the selection of relevant images.

These are described in detail in the next sections.

3.1. Estimation of Local Approximators

Let $X^+ = \{(x'_i, l_i) | l_i = 1, i = 1, 2, \ldots, n_M\}$ be a set of samples obtained from the training data set $T$. In order to construct the local approximators $\phi(\cdot)$, the ARBFN uses these positive samples to estimate the RBF centers and widths. We assign each positive sample to the local approximator $\phi(\cdot)$, so that the shape of each relevant cluster can be described by:

$$\phi(||x - z_i||) = \exp\left(\frac{||x - z_i||^2}{2\sigma_i^2}\right)$$

where the center $z_i, i = 1, 2, \ldots, n_M$ are characterized by the positive samples,

$$\{z_i\} = \{x'_i\}, i = 1, 2, \ldots, n_M$$

The RBF width $\sigma_i$ used in Eq. (7) is estimated by

$$\sigma_i = 0.5 \min_j (||x'_i - x'_j||), \forall x'_j \in X^+, j \neq i$$

Here, only the positive samples are assigned as the centers of the corresponding RBF functions. To take into account the effects of the negative samples, we proposed shifting centers (described in Section 3.3). Hence, the estimated model function $f(x)$ is given by:

$$f(x) = \sum_{i=1}^{n_M} \lambda_i \phi_i(||x - z_i||)$$

The linear weight, $\lambda_i = 1$, indicates that we take all the centers (or the positive samples) into consideration. However, the degree of importance of $\phi_i(\cdot)$ is indicated by the natural responses of the Gaussian-shaped RBF functions and the superposition of the functions.

3.2. A Weighted Norm

The basic RBF version of the ARBFN discussed above (Eq.(7)) is based on the assumption that the feature space is uniformly weighted in all directions. However, image feature variables tend
to exhibit different degrees of importance which heavily depend on the nature of the query and the relevant images defined [1]. This leads to the adoption of an elliptic basis function (EBF):

$$\psi(x, z_i) = \| x - z_i \|^2_\Lambda = (x - z_i)^T \Lambda (x - z_i)$$  \hspace{1cm} (11)

where $\Lambda = diag[\alpha_1, ..., \alpha_p, ..., \alpha_P]$. So, the parameters $\alpha_p, p = 1, ..., P$ represent the relevance weights which are derived from the variance of the positive samples in $X^+$ as follows:

$$\alpha_p = \begin{cases} 1 & \xi_p = 0 \\ 1/\xi_p & o.w. \end{cases}$$  \hspace{1cm} (12)

$$\xi_p = \left( \frac{1}{n_M - 1} \sum_{i=1}^{n_M} (x_{ip} - x_p^*)^2 \right)^{1/2}$$  \hspace{1cm} (13)

The matrix $\Lambda$ is a symmetric matrix, whose diagonal elements $\alpha_p$ assign a specific weight to each input coordinate, determining the degree of the relevance of the features [1]. The adjustable metric described in Eq. (11) is very important, since it permits a more specific dissimilarity measure between inputs and centers than Euclidean metric.

As a result, a new version of the Gaussian-shaped RBF Eq. (7), that takes into account the feature relevancy, can be defined as:

$$\phi(||x - z_i||_\Lambda) = \exp\left( -\frac{\psi(x, z_i)}{2\sigma_i^2} \right)$$  \hspace{1cm} (14)

where $\psi(x, z_i)$ is as given in Eq. (11).

### 3.3. Shifting Centers

The possibility of moving the expansion centers is useful for improving the representativeness of the centers. Recall that, in a given training set, we have both positive and negative samples, which are the ranked results from the previous search operation. Let $X^- = \{ (x^n_i, l_i) \} \subseteq \mathbb{R}^d$, $i = 0, 1, 2, ..., n_\Lambda$ be the set of negative samples. For all negative samples in this set, the similarity scores from the previous search indicate that their clusters are close to the positive samples retrieved. Here, the use of negative samples becomes essential, as the RBF centers should be moved slightly away from these clusters. Therefore, we modify each node vector in the positive sample set, $X^+$, before it is used for characterizing the RBF centers in Eq. (8).

Let the input vector $x''$ (randomly selected from the negative data set) be the closest point to $x'_i$, such that:

$$D_i < D_j; \quad \forall i \in \{1, 2, ..., n_M\}, \quad i \neq i'$$  \hspace{1cm} (15)

where $D_i$ is the weighted distance:

$$D_i = \sum_{p=1}^{P} \alpha_p (x''_p - x'_p)^2$$  \hspace{1cm} (16)

Then, the node vector is modified by the anti-reinforced learning rule:

$$x''_i, (n + 1) = x''_i, (n) - \eta(n) \left[ x''_i, (n) - x'_i, (n) \right]$$  \hspace{1cm} (17)

where $\eta(n)$ is a learning constant which decrease monotonically with the number of iteration $n$, $0 < \eta(n) < 1$. The algorithm is repeated by going through the collection of samples $\{ x''_i, i = 1, 2, ..., n_\Lambda \}$.

### 4. EXPERIMENTAL RESULTS

In this section, we tested all algorithms discussed with the image database from Corel Gallery 65000 product [8]. This contains 40,000 real-life photographs, organized into 400 categories by Corel professionals. These categories were used as a ground truth in our evaluation. For indexing purposes, each image is characterized by visual descriptors using multiple types of features, $F = \{ F_{color}, F_{feature}, F_{shape} \}$ where the representations are color histogram and color moments for color descriptors; Gabor wavelet transform for texture descriptors [9]; and Fourier descriptor for shape descriptors [10]. The feature database obtained was scaled to remove unequal dynamic ranges of each feature variable. We will begin this section by implementing the LMN using the proposed ARBFN architecture and comparing its performance with two other learning strategies. We will then examine the ARBFN and the single-class models discussed in Section 2.

#### 4.1. Retrievals with Local Model Networks

Our proposal in this section is to verify that the proposed ARBFN is able to meet the demands of interactive retrieval application; in particular, where there is a small set of training samples with a high level of correlation between the samples. We compared the ARBFN method with two learning strategies that have been successfully used in other situations to construct the RBF network. Both learning strategies have been implemented and are available from the Neural Network Toolbox using MATLAB [7]. The first learning method, the orthogonal least square (OLS) learning procedure described in [6], was used to identify a RBF network model. The RBF centers were selected according to the network's mean square error criterion. In the second learning method, each vector in a retrieved set was associated with the RBF centers, using a one-to-one correspondence. The weight and bias of the second layers were calculated in such a way that the network sum-squared error was minimized to zero on the training vectors. We denote this method as EDLS (Exact Design Network using Least Squares criterion). The RBF widths for both learning methods were determined experimentally, and it was found that the appropriate width was $\sigma = 0.8$.

We chose 35 images as queries, from different categories. For each query, relevance feedback (subjected to the ground truth classes) was provided on the 16 top-ranked images retrieved. Precision ($Pr$) was recorded after each query iteration.

Table 1 summarizes the average precision results, $\bar{Pr}(t)$, as a function of iteration $t$, taken over the 35 test queries. We can see from the results that the ARBFN significantly improved the retrieval accuracy (up to 92% precision). The first iteration showed an improvement of about 35.9%; the second iteration an additional 9.6%, and the third increased by 2.1%. The ARBFN outperformed the OLS (76.61%) and the EDLS. This result confirms that the ARBFN learning strategy offers a better solution to the construction of a RBF network for the interactive image retrieval, than the two standard learning strategies. In addition, it was observed that the performance of the EDLS reduced after two iterations as the retrieved samples became correlated more strongly. This indicated that the RBF centers critically influenced the performance of the RBF classifier, and that RBF classifier constructed by matching all retrieved samples exactly to the RBF centers degraded the retrieval performance.
4.2. A Comparison to the Single Model

We next compared the learning performance of the ARBFN to the single-class based learning methods discussed in Section 2. This included single-RBF (Eq. (4)) [3], OPT-RF [2], and MARS (Eq. (2)-(3)) [1]. We employ a precision versus recall graph as the criterion from performance measures, using the query set containing 59 images randomly selected from different categories. For every query, the relevance feedback process was done using the top sixteen retrieved images.

Fig. 2 illustrate the average precision versus recall figures after one interaction. This is perhaps the most important precision results, since users would likely provide only one round of relevance feedback. The behavior of the system without learning and the strong improvements with interactive learning can easily be seen. In this condition, the ARBFN achieved the best precision results compared to the other methods discussed. In addition, a comparison of the numerical results after three rounds of relevance feedback shows that the single-RBF needs one and two more rounds to match the first-round precision of the ARBFN, whereas the OPT-RF needed at least two more rounds to match the first-round precision of the ARBFN.

Fig. 3 graphically illustrates the learning performances of the ARBFN compared with the single-RBF, for the query “SWIMMERS”. This clearly indicates the superiority of the proposed learning approach.

![Average precision versus recall](image)

**Fig. 2.** Average precision versus recall figure (after first round of relevance feedback), obtained by retrieving 59 queries, using Corel database. Note that, results obtained by simple-CBIR are used as a benchmark for other interactive methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>$t=0$</th>
<th>$t=1$</th>
<th>$t=2$</th>
<th>$t=3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARBFN</td>
<td>44.82</td>
<td>80.72</td>
<td>90.36</td>
<td>92.50</td>
</tr>
<tr>
<td>EDLS</td>
<td>44.82</td>
<td>50.18</td>
<td>43.39</td>
<td>43.04</td>
</tr>
<tr>
<td>OLS</td>
<td>44.82</td>
<td>66.07</td>
<td>73.21</td>
<td>76.61</td>
</tr>
</tbody>
</table>

**Table 1.** Average precision (%) as a function of iteration, $P_r(t)$, obtained by retrieving 35 queries, using Corel database.

![Retrieval result](image)

**Fig. 3.** Retrieval result of the query “SWIMMERS”, obtained by (a) simple-CBIR; (b) single-RBF; (c) ARBFN, based on the one-round of relevance feedback.

6. REFERENCES


