

# GAUSSIAN MIXTURE MODEL FOR RELEVANCE FEEDBACK IN IMAGE RETRIEVAL\*

*Fang Qian<sup>1</sup>, Mingjing Li<sup>2</sup>, Lei Zhang<sup>2</sup>, Hong-Jiang Zhang<sup>2</sup>, Bo Zhang<sup>1</sup>*

<sup>1</sup>Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

<sup>2</sup>Microsoft Research Asia, 49 Zhichun Road, Beijing 100080, China

## ABSTRACT

Relevance Feedback (RF) has become a powerful technique in content-based image retrieval. Most RF methods assume that positive images follow the single Gaussian distribution, which is not sufficient to model the actual distribution of images due to the gap between the semantic concept and low-level features. In this paper, Gaussian mixture model (GMM) is applied to represent the distribution of positive images in relevance feedback, and a novel method is proposed to estimate the parameters of GMM. Both positive and negative examples are used to estimate the number of Gaussian components. Furthermore, due to the lack of training samples, unlabeled data are also incorporated to estimate the covariance matrices. Experimental results show that our GMM-based RF method outperforms that based on single Gaussian model.

## 1. INTRODUCTION

Efficient image browsing and retrieval tools can help people utilizing their digital image collections. A number of general-purpose image search engines have been developed both in commercial and research area. The common approach of early systems is to represent each image as a feature vector and to assess similarity between images by a metric. Many visual features have been explored to describe the color, texture or shape of images. While these work established the basis of content-based image retrieval (CBIR), their usefulness is limited due to the following two reasons: one is the gap between low-level features and high-level concepts; another is the human perception subjectivity. Thus, a fixed feature representation and similarity measure method is hard to satisfy different users. To address these problems, interactive relevance feedback (RF) has been applied to image retrieval [4]. The idea is to let the user guide the retrieval system. During retrieval process, the user interacts with the system and rates the relevance of the

retrieved images, according to his/her subjective judgment. Based on the feedback information, the system dynamically learns the user's intention, and gradually presents better results.

Query point movement and feature re-weighting are two widely-used approaches. The former essentially attempts to improve retrieval results by moving the query vector towards positive examples and away from negative examples, based on the assumption that all positive examples cluster together in feature space. The latter is similar to the idea of "term frequency" and "inverse document frequency" (TF\*IDF) in text retrieval. That is, the system increases the importance of those dimensions of a feature vector that do help in retrieving relevant images and reduce the importance of those dimensions that do not. An intuitive method is to weight the feature by a decrease function of the standard deviation of all relevant images along the axis [4]. The MindReader System [2] integrated the above two independent processes (query movement and feature re-weighting) into a unified framework, based on the minimization of total distances of positive examples from the new query.

Those approaches are based on the assumption that positive samples follow the single Gaussian distribution. However, due to the well-known gap between the semantic concept and low-level feature, the relevant images are usually separated by irrelevant ones and thus cannot be well depicted by a single Gaussian model.

In this paper, we propose to use Gaussian mixture model (GMM) in representing the distribution of relevant images. We propose a simple method to estimate the number of components based on both positive and negative examples. Furthermore, to reduce the problem of lacking training samples, unlabeled data are also incorporated to estimate the covariance matrices of GMM.

The rest of the paper is organized as follows. The related work based on single Gaussian model is reviewed in Section 2. We give a brief description of Gaussian mixture model and present our parameter estimation method in Section 3. Experimental results and conclusions are given in Section 4 and Section 5, respectively.

---

\* This work was performed at Microsoft Research Asia.

## 2. RELATED WORK

### 2.1. The MindReader Approach

Let  $N$  be the number of positive samples and let  $\mathbf{x}_i = [x_{i1}, \dots, x_{id}]^T$  be a  $d$ -dimensional vector that represents  $i$ -th image ( $i=1, \dots, N$ ).  $\mathbf{X}$  denotes an  $N \times d$  matrix  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T$ . Vector  $\mathbf{v} = [v_1, \dots, v_N]^T$  represents the degree of relevance for the  $N$  relevant images given by the user. The proposed distance function by MindReader [2] is

$$D(\mathbf{x}, \mathbf{q}) = (\mathbf{x} - \mathbf{q})^T \mathbf{M}(\mathbf{x} - \mathbf{q}). \quad (1)$$

The optimal solutions to  $\mathbf{q}$  and  $\mathbf{M}$  are as follows:

$$\mathbf{q} = \frac{\mathbf{X}^T \mathbf{v}}{\sum_{i=1}^N v_i}, \quad (2)$$

$$\mathbf{M} = (\det(\mathbf{C}))^{\frac{1}{d}} \mathbf{C}^{-1}, \quad (3)$$

where  $\mathbf{C}$  is the weighted covariance matrix, whose elements are computed in the following way:

$$c_{jk} = \sum_{i=1}^N v_i (x_{ik} - q_k)(x_{ij} - q_j). \quad (4)$$

The optimal query  $\mathbf{q}$  in (2) turns out to be the weighted average of relevant images. The symmetric full matrix  $\mathbf{M}$  enables the MindReader system to guess ‘‘diagonal queries’’. This approach is essentially based on single Gaussian distribution model.

### 2.2. Probability-Based Approach

Probabilistic methods have been applied to image retrieval [5, 6]. Assuming all the positive samples obey Gaussian distribution; then the probability density function of  $\mathbf{x}$  is

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right\}, \quad (5)$$

where the mean vector  $\boldsymbol{\mu}$  and the covariance matrix  $\boldsymbol{\Sigma} = [\sigma_{jk}]$  can be computed using  $N$  positive samples:

$$\boldsymbol{\mu} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i, \quad (6)$$

$$\sigma_{jk}^2 = \frac{1}{N} \sum_{i=1}^N (x_{ik} - \mu_k)(x_{ij} - \mu_j), \quad (1 \leq j, k \leq d). \quad (7)$$

The rank order of  $\mathbf{x}$  is determined by  $p(\mathbf{x})$ . It can be easily derived from Formula (5) that

$$\text{Rank}(p(\mathbf{x})) = \text{Rank}\left(-(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right). \quad (8)$$

This result is consistent with the MindReader system when all positive examples have the same degree of relevance.

### 2.3. The MARS Approach

Assuming that all of the features are independent of each

other, the covariance matrix  $\boldsymbol{\Sigma}$  is simplified to a diagonal matrix:

$$\boldsymbol{\Sigma} = \text{diag}(\sigma_{11}^2, \sigma_{22}^2, \dots, \sigma_{dd}^2), \quad (9)$$

where

$$\sigma_{ij} = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_{ij} - \mu_j)^2}. \quad (10)$$

Therefore, (8) can be simplified to:

$$\text{Rank}(p(\mathbf{x})) = \text{Rank}\left(\sum_{i=1}^d -\left(\frac{x_i - \mu_i}{\sigma_{ii}}\right)^2\right). \quad (11)$$

That is, when all of the features are assumed to be independent of each other, the probability-based method is equivalent to the query point movement and feature re-weighting method in MARS [4], which is a special case of MindReader. The new query is the mean vector of all positive samples. The  $i$ -th feature is weighted by  $1/\sigma_{ii}$ , where  $\sigma_{ii}$  is the standard deviation of all relevant images along the  $i$ -th axis.

### 2.4. Discussion

The Gaussian distribution assumption is reasonable for those relevant images around the query in a local area, and the above approaches do improve the performance of retrieval. However, as the feedback process goes on, more positive samples far away from the query can be retrieved, and the single Gaussian density function is no longer sufficient to model their distribution. This problem stems from the gap between low-level features and high-level concept. That is, images sharing same semantic concept are often separated by irrelevant images in low-level feature space. Therefore, the single Gaussian distribution assumption for positive samples will limit the further improvement of retrieval performance.

## 3. RELEVANCE FEEDBACK BASED ON GAUSSIAN MIXTURE MODEL

### 3.1. Gaussian Mixture Model

Mixture models are a type of density models which that comprise a number of component functions, usually Gaussian. A mixture of  $K$  Gaussians is:

$$\begin{aligned} p(\omega|x) &= \sum_{k=1}^K \alpha_k G(x; \mu_k, \sigma_k) \\ &= \sum_{k=1}^K \alpha_k \cdot \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k)\right\}, \quad (12) \end{aligned}$$

where  $\alpha_k$  is the mixing parameter satisfying  $\sum_{k=1}^K \alpha_k = 1$ .

$G(x; \mu_k, \sigma_k)$  is the probability density function (p.d.f.)

corresponding to the  $k$ -th Gaussian component. The Gaussian mixture model contains the following adjustable parameters:  $\alpha_i, \mu_i$  and  $\Sigma_i$  ( $i=1, \dots, K$ ).

Parameter estimation in GMM is nontrivial. Many works used Expectation-Maximisation (EM) algorithm [6], which is a well established maximum likelihood algorithm for fitting a mixture model to a set of training data [1]. It should be noted that EM requires *a priori* selection of the number of  $K$  components, which is practically impossible in the retrieval / feedback process. Furthermore, EM algorithm is time-consuming due to its iterative nature, whereas real-time processing is necessary to relevance feedback. To overcome the difficulties, we propose a simple but effective method to estimate the number of components, and the parameters of each component.

### 3.2. Estimating the Number of Gaussian Components

Zhang et al. proposed an image retrieval method based on a set of coverings [8]. A covering is a hyper sphere in the feature space, which tries to contain as many positive examples as possible, but none of negative examples. This implies that positive examples may be grouped in the feature space and these groups are separated by negative examples. Thus, an intuitive idea is to use the number of coverings as the estimate of the number of Gaussian components.

Both positive and negative examples are used to estimate the number of coverings. In the extreme case with no negative examples, this number is one. Let  $\mathbf{I}^+ = \{I_1^+, I_2^+, \dots, I_{N^+}^+\}$  represent the set of positive examples, and  $\mathbf{I}^- = \{I_1^-, I_2^-, \dots, I_{N^-}^-\}$  represent the set of negative examples. Note that the initial query is treated as a positive image in  $\mathbf{I}^+$  by default. The aim is to find a set of coverings  $\mathbf{C}^+ = \{(C_k^+, R_k^+), k=1, \dots, K\}$  to cover  $\mathbf{I}^+$ , where  $C_k^+$  and  $R_k^+$  is the center and radius of the  $k$ -th covering.

If  $\mathbf{I}^+$  is not empty, a covering can be constructed in the following way. First, identify the image  $I_i^+$  in  $\mathbf{I}^+$  with the largest likelihood of being relevant, and set the center of new covering to  $I_i^+$ . Next, calculate the maximal distance between  $I_i^+$  and  $\mathbf{I}^+$ , and the minimal distance between  $I_i^+$  and  $\mathbf{I}^-$ , i.e.:

$$d_{\max} = \max_j D(I_i^+, I_j^+), \quad \forall I_j^+ \in \mathbf{I}^+ \quad (13)$$

$$d_{\min} = \min_j D(I_i^+, I_j^-), \quad \forall I_j^- \in \mathbf{I}^- \quad (14)$$

Then the radius of covering can be determined as follows:

$$r = \begin{cases} (d_{\min} + d_{\max})/2, & \text{if } d_{\min} \geq d_{\max} \\ \rho \cdot d_{\min}, & \text{otherwise} \end{cases}, \quad (15)$$

where  $\rho$  is a constant satisfying  $0 < \rho < 1$ . It is set to 0.95 in our experiment. Last, positive examples in  $\mathbf{I}^+$  that fall in this covering are removed from  $\mathbf{I}^+$  and assigned to the covering.

This process is repeated until all positive examples have been covered. The resulting number of coverings is the number of Gaussian components, and the examples in each covering are used to estimate the parameters of each component.

### 3.3. Estimating the Mean and Covariance Matrix

Obviously, the mean vector of each Gaussian component can be obtained by averaging all positive samples in the corresponding covering:

$$\boldsymbol{\mu}_k = \frac{1}{m_k} \sum_{i=1}^{m_k} \mathbf{x}_i^{(k)}, \quad k=1, \dots, K \quad (16)$$

where  $m_k$  is the number of positive samples belong to the  $k$ -th covering.

However, it is not easy to estimate the covariance matrix due to insufficient number of training samples. A full covariance matrix needs to estimate  $d(d+1)/2$  elements while the number of samples is usually  $O(d)$ . For simplicity, we assume that all dimensions of the feature vector are independent of each other. Thus the covariance matrix  $\boldsymbol{\Sigma}_k$  is simplified to a diagonal matrix. Furthermore, we use non-feedback images together with the positive examples in parameter estimation. Combining unlabeled data with labeled one has been showed to be useful for making up the small sample problem in training [3][7].

We assume those unlabeled data falling in the coverings are relevant images, and use them to help estimate the covariance matrices:

$$\sigma_{ij}^{(k)} = \sqrt{\frac{1}{N_k} \sum_{i \in C_k} (x_{ij} - \mu_j)^2}, \quad (17)$$

where  $N_k$  is the total number of positive and unlabeled images falling in covering  $C_k$ . Note that  $N_k$  is greater or equal to  $m_k$  in Formula (16).

If a covering contains only one positive sample but not any unlabeled image, we use all positive examples to estimate the covariance matrix of this component.

### 3.4. Estimating the Weight of Each Component

The weight of the  $k$ -th component is set to the portion of positive examples falling in the corresponding covering:

$$\alpha_k = m_k / N^+, \quad (18)$$

where  $m_k$  is the number of positive examples in the  $k$ -th covering, and  $N^+$  is the total number of positive examples.

### 3.5. Relevance Feedback Method

When feedback images are available, the parameters of a Gaussian mixture model are estimated by the above mentioned methods. For each image in the database, the probability of being relevant is calculated according to (12). Then images with the highest probabilities are returned to the user as the refined retrieval result.

## 4. EXPERIMENTS

In the experiments, 10,000 images from Corel data set are used to form the image database, while 100 images from ten categories are used to form the query image set. These ten categories include butterfly, eagle, elephant, flower, forest, fungus, leopard, sunset, tiger, and waterfall. Image from the same category as that of the query are used as the ground truth.

Relevance feedback is conducted automatically. In the first iteration of feedback, top 30 images are checked and labeled as either positive or negative examples. In the following iterations, the labeled positive images are ranked at beginning, while the negative images are ranked at last, and top 30 unlabeled images are checked. In each round, all positive and negative images available are used to estimate the parameters.

Color moments and wavelet based texture feature are used. The first two moments (mean and standard deviation) are extracted from the three color channels (HSV space) and therefore form a 6-dimensional feature vector. For wavelet based texture, the original image is decomposed into 10 de-correlated sub-bands through 3-level wavelet transform. In each sub-band, the standard deviation of the wavelet coefficients is extracted, resulting in a 10-dimensional feature vector.

We have compared our GMM-based RF approach with MARS approach which is based on single Gaussian model. The performance measure is precision at scope 100. Five iterations of feedback were tested.

In the first round of RF, the Gaussian mixture model performs nearly as same as single Gaussian model. As the relevance feedback goes on, the mixture Gaussian model outperforms the single Gaussian model. This can be illustrated as follows: at first, few positive samples can be retrieved, and they are all around the query in the feature space. Thus the Gaussian model is still suitable to describe the distribution of relevant images. With the feedback going on, positive samples become more sufficient and different, the Gaussian mixture model become more suitable to characterize the distribution.

## 5. CONCLUSIONS

In this paper, we had proposed a relevance feedback method based on Gaussian mixture model. It is more

reasonable to assume that positive examples follow Gaussian mixture model instead of single Gaussian distribution. We proposed the covering method to estimate the number of Gaussian components, and we used the unlabeled images to help estimate the parameters. The experimental results demonstrate the effectiveness of the proposed approach.

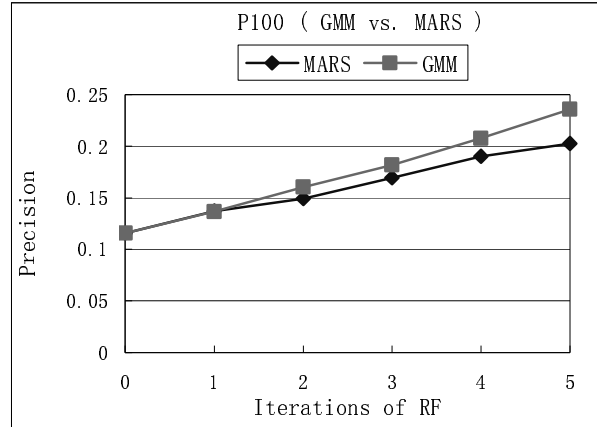


Figure 1. Retrieval performance comparison

## 6. REFERENCES

- [1] J. A. Bilmes, "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models", Technical Report, University of Berkeley, ICSI-TR-97-021, 1997.
- [2] Y. Ishikawa and R. Subramanya, "MindReader: Query databases through multiple examples", in Proc. of the 24<sup>th</sup> VLDB conference, (New York), 1998.
- [3] T. Joachims, "Transductive Inference for Text Classification using Support Vector Machines", Proc. 16th International Conference on Machine Learning, pp. 200-209, San Francisco, CA, USA, 1999. Morgan Kaufmann.
- [4] Y. Rui, T. S. Huang, M. Ortega and S. Mehrotra, "Relevance feedback: A power tool in interactive content-based image retrieval", IEEE Transaction on Circuits and Systems for Video Technology, Special Issue on Segmentation, Description, and Retrieval of Video Content, 8(5): 644-655, September 1998.
- [5] Z. Su, S. Li, H. J. Zhang, "Extraction of Feature Subspaces for Content-Based Retrieval Using Relevance Feedback", in Proc. ACM Multimedia 2001, Ottawa, Canada, Sept. 2001
- [6] N. Vasconcelos and A. Lippman, "Embedded Mixture Modeling for Efficient Probabilistic Content-Based Indexing and Retrieval", in Proc. of SPIE Conf. on Multimedia Storage and Archiving Systems III, Boston, 1998.
- [7] Y. Wu, Q. Tian, and T. S. Huang, "Discriminant-EM Algorithm with Application to Image Retrieval", in Proc. IEEE Conf. Computer Vision and Pattern Recognition, South Carolina, June, 2000.
- [8] L. Zhang, F. Z. Lin, and B. Zhang, "A neural network based self-learning algorithm of image retrieval", Chinese Journal of Software, 12(10): 1479-1485, October 2001.