

Gaussian Mixture Distance for Information Retrieval

X.Q. Li and I. King
{xqli, king}@cse.cuhk.edu.hk
Department of Computer Science & Engineering
The Chinese University of Hong Kong
Shatin, New Territories, Hong Kong

Abstract

In most Information Retrieval (IR) applications, Euclidean distance is used for similarity measurement. It is adequate in many cases but this distance metric is not very accurate when there exist some different local data distributions in the database. We propose a Gaussian mixture distance for performing accurate nearest-neighbor search for Information Retrieval (IR). Under an established Gaussian finite mixture model for the distribution of the data in the database, the Gaussian mixture distance is formulated based on minimizing the Kullback-Leibler (KL) divergence between the distribution of the retrieval data and the data in database. We compared the performance of the Gaussian mixture distance with the well-known Euclidean and Mahalanobis distance based on a precision performance measurement. Experimental results demonstrate that the Gaussian mixture distance function is superior in the others for different types of testing data.

1 Introduction

Research on Information Retrieval (IR) is a key area for building and managing large multimedia databases. Given a query, the objective of IR is to find out the most relevant data of the query according to a similarity metric. To describe the similarity, a form of the geometrical distance metric between points in the feature space is considered a logical choice in many IR applications. The smaller the distance between two vectors in the feature space, the greater the similarity. Thus, the nearest-neighbor search technique is often employed to retrieve the most similar data in the database corresponding to the a query.

A general definition of the distance between two d -dimensional feature vectors \vec{q} and \vec{x} is

$$D = \|\vec{q} - \vec{x}\|_{\mathcal{A}}^2 = (\vec{q} - \vec{x})^T \mathcal{A}^{-1} (\vec{q} - \vec{x}), \quad (1)$$

where we call the matrix, \mathcal{A} , as the distance matrix which is

any positive definite $d \times d$ matrix. The effect of \mathcal{A} is to scale the distance along each feature axis.

In practice, the most two commonly used distance measures in IR are (1) the Euclidean distance whose distance matrix is the identity matrix and (2) the Mahalanobis distance whose distance matrix is the inverse covariance matrix of the data objects. The main weakness of the above two metrics is that they are not accurate when there exist some clusters with different distributions in the feature space since (1) the Euclidean distance is independent of the distribution of the data, and (2) the Mahalanobis distance only considers the global distribution of the data. In our research, to overcome the weakness, we formulate a Gaussian mixture distance according to a Gaussian finite mixture model which estimates the underlying local distribution of the database. A similar Gaussian mixture distance has been used by Cox *et al.* for face recognition with some satisfactory results [1]. In this paper, we extend Cox's work using the KL divergence and a Gaussian mixture model together to formulate an optimal metric under the accuracy consideration for similarity measure when the estimation of data has been established. In our research, we find the Gaussian mixture distance is a more accurate similarity measure than the Euclidean and the Mahalanobis distance since it captures the local distribution of the database.

The Gaussian mixture distance is highly dependent on the estimation of the distribution of the data. To establish an optimal mixture model for the data in database, the "actual" number of the mixtures in the Gaussian mixture model needs to be determined. In our research, we select a mixture number determination criterion based on the Bayesian YING-YANG (BYY) theory for its robust performance [2, 3].

In the next section, we formulate the Gaussian mixture distance in detail using the KL divergence method. Section 3 introduces a mixture number determination criterion based on BYY theory to obtain the optimal mixture numbers of the Gaussian mixture model using EM algorithm. In section 4, we demonstrate our experimental results when comparing the performance of using different distance definitions

for the nearest-neighbor search using a synthetically generated database. Finally, a short conclusion is given in Section 5.

2 Gaussian Mixture Distance Function

2.1 Probability Model for Nearest-Neighbor Search

Given a set of d -dimensional feature vectors, $\mathcal{X} = \{\vec{x}_i\}_{i=1}^N$, and a query, \vec{q} , the objective of IR is to find out a subset \mathcal{C} which contains M retrieved data with the M greatest values of the conditional probability density $p(\vec{x}_i|\vec{q})$, where $\mathcal{C} \subseteq \mathcal{X}$ and $p(\vec{x}_i|\vec{q})$ is a retrieval probability function. If we estimate the probability density function $p(\vec{x}|\vec{q})$, as a Gaussian distribution whose mean is the query vector \vec{q} , then

$$p(\vec{x}|\vec{q}) = \frac{1}{(2\pi)^{d/2}|\Sigma_{\mathcal{C}}|^{1/2}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{q})^T \Sigma_{\mathcal{C}}^{-1}(\vec{x} - \vec{q})\right), \quad (2)$$

where $\Sigma_{\mathcal{C}}$ is the covariance matrix and $|\Sigma_{\mathcal{C}}|$ is the determinant of $\Sigma_{\mathcal{C}}$. Notice that

$$\max_{\vec{x}_i} p(\vec{x}_i|\vec{q}) \propto \max_{\vec{x}_i} \log p(\vec{x}_i|\vec{q}),$$

where

$$\begin{aligned} \log p(\vec{x}_i|\vec{q}) &= -\log\left((2\pi)^{d/2}|\Sigma_{\mathcal{C}}|^{1/2}\right) \\ &\quad -\frac{1}{2}(\vec{x}_i - \vec{q})^T \Sigma_{\mathcal{C}}^{-1}(\vec{x}_i - \vec{q}). \end{aligned} \quad (3)$$

Hence,

$$\begin{aligned} \max_{\vec{x}_i} p(\vec{x}_i|\vec{q}) &\propto \min_{\vec{x}_i} (\beta(\vec{x}_i - \vec{q})^T \Sigma_{\mathcal{C}}^{-1}(\vec{x}_i - \vec{q})) \\ &= \min_{\vec{x}_i} (\vec{x}_i - \vec{q})^T [\beta \Sigma_{\mathcal{C}}^{-1}] (\vec{x}_i - \vec{q}), \end{aligned} \quad (4)$$

where β is a positive constant. In Eqn. (4),

$$(\vec{x}_i - \vec{q})^T [\beta \Sigma_{\mathcal{C}}^{-1}] (\vec{x}_i - \vec{q}) \quad (5)$$

can be considered as a distance function and the task of retrieval becomes to search M nearest-neighbors of \vec{q} in the database based on the distance function, where $\beta \Sigma_{\mathcal{C}}^{-1}$ can be seen as the distance matrix and $\Sigma_{\mathcal{C}}^{-1}$ is the major parameter needs to be estimated.

2.2 Distance Matrix under a Single Gaussian Model

In practice, the identity matrix \mathcal{I} is one of the most common choices as the distance matrix, and therefor Eqn. (5)

becomes the Euclidean distance, where $\Sigma_{\mathcal{C}}^{-1}$ is estimated as $\frac{1}{\beta} \mathcal{I}$ which is a diagonal matrix. Another popular choice of $\Sigma_{\mathcal{C}}^{-1}$ is the inverse covariance matrix of the database, $\Sigma_{\mathcal{C}}^{-1} \equiv \Sigma_{\mathcal{X}}^{-1}$, where $\Sigma_{\mathcal{X}}$ is the covariance matrix of the whole database. Then Eqn. (4) becomes the well-known Mahalanobis distance. However, above choices of the distance matrix mainly depend on researchers' experience or intuition. Hence, a theoretical criterion for distance metric estimation is required. Next, we give a method for estimating the distance matrix based on KL divergence [4].

Let $p(\vec{x})$ denotes a probability density function (pdf) of the feature vector \vec{x} to represent the distribution of the data in the database. Since the retrieved data must derive from the database, an important criterion for estimating $p(\vec{x}|\vec{q})$ is to minimize the divergence between $p(\vec{x}|\vec{q})$ and $p(\vec{x})$. Here, we use the relative entropy(KL divergence) to describe the divergence,

$$KL(p(\vec{x}|\vec{q})||p(\vec{x})) = \int p(\vec{x}|\vec{q}) \log\left(\frac{p(\vec{x}|\vec{q})}{p(\vec{x})}\right) d\vec{x}. \quad (6)$$

Hence, the optimal estimation, $\hat{\Sigma}_{\mathcal{C}}$, of $\Sigma_{\mathcal{C}}$ is

$$\hat{\Sigma}_{\mathcal{C}} \equiv \arg \min_{\Sigma_{\mathcal{C}}} KL(p(\vec{x}|\vec{q})||p(\vec{x})). \quad (7)$$

If $p(\vec{x})$ is a Gaussian distribution with the covariance matrix $\Sigma_{\mathcal{X}}$ and the mean \vec{m} , the KL divergence becomes,

$$\begin{aligned} KL(p(\vec{x}|\vec{q})||p(\vec{x})) &= \frac{1}{2} \log \frac{|\Sigma_{\mathcal{X}}|}{|\Sigma_{\mathcal{C}}|} + \frac{1}{2} tr [\Sigma_{\mathcal{C}}(\Sigma_{\mathcal{X}}^{-1} - \Sigma_{\mathcal{C}}^{-1})] \\ &\quad + \frac{1}{2} tr [\Sigma_{\mathcal{X}}^{-1}(\vec{m} - \vec{q})(\vec{m} - \vec{q})^T], \end{aligned} \quad (8)$$

where $tr[\cdot]$ denotes the trace of the matrices. The KL divergence has the property that it is non-negative, $KL(p(\vec{x}|\vec{q})||p(\vec{x})) \geq 0$. Let $\frac{dKL}{d\Sigma_{\mathcal{C}}}$ denotes the partial derivatives of KL to each element in $\Sigma_{\mathcal{C}}$, then,

$$\frac{dKL}{d\Sigma_{\mathcal{C}}} = \frac{1}{2} tr [(\Sigma_{\mathcal{X}}^{-1} - \Sigma_{\mathcal{C}}^{-1}) [1]_{k \times k}]. \quad (9)$$

Hence, the optimal estimate of $\Sigma_{\mathcal{C}}$ is

$$\hat{\Sigma}_{\mathcal{C}} \equiv \arg \min_{\Sigma_{\mathcal{C}}} KL(p(\vec{x}|\vec{q})||p(\vec{x})) = \Sigma_{\mathcal{X}}. \quad (10)$$

From the above results, the Mahalanobis distance using $\Sigma_{\mathcal{X}}^{-1}$ can be considered as an optimal metric for estimating $\Sigma_{\mathcal{C}}$ when the distribution of all data in the database can be well described using a single Gaussian. And especially when the Gaussian is a norm Gaussian, $\Sigma_{\mathcal{X}} = a\mathcal{I}$, where a is a positive constant and \mathcal{I} is the identity matrix, the distance can be seen as an Euclidean distance which is a special case of Mahalanobis distance. Fig. 1 shows cases of using the Mahalanobis distance and Euclidean distance. The two distance matrices are uniform and independent of the position of the query \vec{q} .

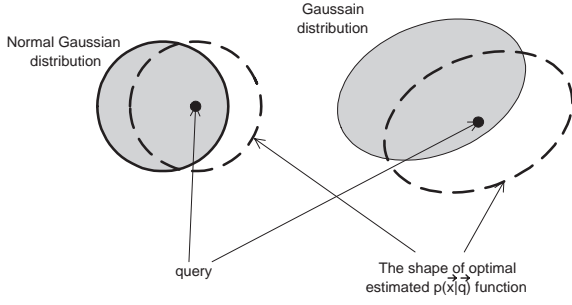


Figure 1: The distributions where the Euclidean and Mahalanobis distance are the optimal metrics

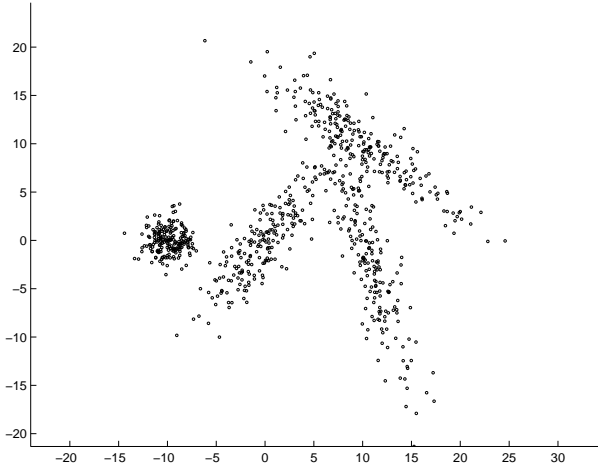


Figure 2: Some clusters in a data set with different local distributions.

Next, let us consider a data set which consists of some different types of data clustered as illustrated in Fig. 2. It becomes unsuitable to use a uniform distance matrix for the different queries in this case. Here, we introduce a method to estimate the distance matrix when using a Gaussian mixture model for data clustering. We call the distance function as the *Gaussian mixture distance*.

2.3 Distance Matrix under a Gaussian Mixture Model

The finite Gaussian mixture model has been widely accepted for data clustering since the Expectation-Maximum (EM) algorithm was introduced by Dempster *et al.* in [5]. This mixture model estimates the data distribution as the sum of k weighted Gaussian distributions,

$$p(\vec{x}) = \sum_{j=1}^k \alpha_j G(\vec{x}, \vec{m}_j, \Sigma_{\mathcal{X}_j}), \quad (11)$$

where each weight, $\alpha_j \geq 0$ and $\sum_{j=1}^k \alpha_j = 1$, and each $G(\vec{x}, \vec{m}_j, \Sigma_{\mathcal{X}_j})$ is a single Gaussian function with the mean, \vec{m}_j and the covariance matrix, $\Sigma_{\mathcal{X}_j}$. In the mixture model, an “incomplete data”, j , which represents the index cluster number is introduced and the conditional probability $P(j|\vec{x})$ ¹ describes the probability of a feature vector \vec{x} belonging to the j -th cluster. Hence, we can use a joint probability density function, $p(\vec{x}, j) = p(\vec{x}|j)P(j)$, of \vec{x} and j to describe the distribution in the database, where $P(j) = \alpha_j$ and $p(\vec{x}|j) = G(\vec{x}, \vec{m}_j, \Sigma_{\mathcal{X}_j})$.

Following the above definitions, we estimate the distribution of retrieved data \mathcal{C} as $p(\vec{x}|\vec{q}) = \sum_{j=1}^k P(j|\vec{q})p(\vec{x}|\vec{q}, j)$, where $p(\vec{x}|\vec{q})$ is a Gaussian function whose mean is the query vector \vec{q} and covariance matrix is $\Sigma_{\mathcal{C}}$. Hence, the KL divergence between the distribution of the retrieved data and the data in database is

$$\begin{aligned} KL(p(\vec{x}, j|\vec{q})||p(\vec{x}, j)) \\ &= \int \int p(\vec{x}, j|\vec{q}) \log \left(\frac{p(\vec{x}, j|\vec{q})}{p(\vec{x}, j)} \right) d\vec{x} dj \\ &= KL(p(j|\vec{q})||p(j)) + KL(p(\vec{x}|j, \vec{q})||p(\vec{x}|j)). \end{aligned} \quad (12)$$

In Eqn. (12),

$$KL(p(j|\vec{q})||p(j)) = \sum_{j=1}^k P(j|\vec{q}) \log \left(\frac{P(j|\vec{q})}{P(j)} \right), \quad (13)$$

and

$$\begin{aligned} KL(p(\vec{x}|j, \vec{q})||p(\vec{x}|j)) \\ &= \sum_{j=1}^k P(j|\vec{q}) \int p(\vec{x}|\vec{q}) \log \left(\frac{p(\vec{x}|\vec{q})}{G(\vec{x}, \vec{m}_j, \Sigma_{\mathcal{X}_j})} \right) d\vec{x}. \end{aligned} \quad (14)$$

Then,

$$\frac{dKL}{d\Sigma_{\mathcal{C}}} = \frac{1}{2} tr \left[\left(\sum_{j=1}^k P(j|\vec{q}) \Sigma_{\mathcal{X}_j}^{-1} - \Sigma_{\mathcal{C}}^{-1} \right) [1]_{k \times k} \right]. \quad (15)$$

Hence, the optimal estimate of $\Sigma_{\mathcal{C}}$ is

$$\hat{\Sigma}_{\mathcal{C}} \equiv \arg \min_{\Sigma_{\mathcal{C}}} KL(p(\vec{x}|\vec{q})||p(\vec{x})) = \left(\sum_{j=1}^k P(j|\vec{q}) \Sigma_{\mathcal{X}_j}^{-1} \right)^{-1}, \quad (16)$$

where

$$P(j|\vec{q}) = \frac{\alpha_j G(\vec{q}, \vec{m}_j, \Sigma_{\mathcal{X}_j})}{\sum_{i=1}^k \alpha_i G(\vec{q}, \vec{m}_i, \Sigma_{\mathcal{X}_i})}. \quad (17)$$

Obviously, the performance of the distance highly depends on the estimation $p(\vec{x})$ of the data in the database. The main

¹In the paper, we use $P(\cdot)$ to denote probability and $p(\cdot)$ to denote probability density. For a discrete variable j , its probability density $p(j)$ can be written as $p(j) = \lim_{\Delta \rightarrow 0} \frac{P(j)}{\Delta}$.

problem concerning with the estimation of $p(\vec{x})$ is model selection, i.e., how to detect the “true” mixture number in the database. Notice that, Mahalanobis distance can be seen a special case of the Gaussian mixture distance where there only exists one mixture in the feature space. In the next section, we will introduce a cluster number selection method based on the BYY theory.

3 Mixture Number Selection

Determine the number of mixtures is a well-known model selection problem that has an important role in unsupervised learning. Recently, a Bayesian-Kullback scheme, called the YING-YANG Learning Theory and System, has been proposed to act as a general learning scheme for unifying existing major unsupervised and supervised learning schemes [2, 3]. One special case of the YING-YANG machine can provide us a criterion for solving the problem of selecting cluster number. In our research, we choose an optimal number of mixtures using the following steps:

Step 1: Set $k \leftarrow 1$, where k is the candidate mixture number.

Step 2: Estimate the parameters, $\Theta = \{\alpha_j, \vec{m}_j, \Sigma_{\mathcal{X}_j}\}_{j=1}^k$, of the Gaussian mixture model under the number of mixtures, k , using the EM algorithm as follows:

E step:

$$P(j|\vec{x}_i) = \frac{\alpha_j^{old} G(\vec{x}_i|j, \vec{m}_j^{old}, \Sigma_{\mathcal{X}_j}^{old})}{\sum_{j=1}^k \alpha_j^{old} G(\vec{x}_i|j, \vec{m}_j^{old}, \Sigma_{\mathcal{X}_j}^{old})}. \quad (18)$$

M step:

$$\begin{aligned} \alpha_j^{new} &= \frac{1}{N} \sum_{i=1}^N P(j|\vec{x}_i), \\ \vec{m}_j^{new} &= \frac{\sum_{i=1}^N P(j|\vec{x}_i) \vec{x}_i}{\sum_{i=1}^N P(j|\vec{x}_i)}, \\ \Sigma_{\mathcal{X}_j}^{new} &= \frac{\sum_{i=1}^N P(j|\vec{x}_i) (\vec{x}_i - \vec{m}_j^{old})(\vec{x}_i - \vec{m}_j^{old})^T}{\sum_{i=1}^N P(j|\vec{x}_i)}. \end{aligned} \quad (19)$$

Step 3: Evaluate the choice of k with the following BYY criterion,

$$J(k) = \frac{1}{2} \sum_{j=1}^k \alpha_j \log |\Sigma_{\mathcal{X}_j}| - \sum_{j=1}^k \alpha_j \log \alpha_j. \quad (20)$$

The Eqn.(20) can be seen as a cost function: the smaller the $J(k)$, the smaller the cost of choosing k as the optimal number of mixtures. As shown in paper [2], the

number selection criterion is that the curve of cost function $J(k)$ versus k reaches its global minimum point at $k = k'$, where k' is the actual number of mixtures in the finite Gaussian mixture.

Step 4: Choose a new $k \leftarrow k + 1$, repeat the step 2 and 3 till k reaches a predefined number.

After using the above steps, we choose the k which minimizes the $J(k)$ as the optimal number of mixtures. Under the optimal estimated k and the parameters $\Theta = \{\alpha_j, \vec{m}_j, \Sigma_{\mathcal{X}_j}\}_{j=1}^k$, we can formulate the Gaussian mixture distance for nearest-neighbor search. Next, we perform some experiments to compare the performance of the Gaussian mixture, the Euclidean and the Mahalanobis distance in the next section.

4 Experiments

4.1 Test Data

We tested the accuracy performance of the Gaussian mixture distance for IR with 3 sets of 2-dimensional synthetic feature vectors. Each test set contains 1200 vectors generated from 6 Gaussian mixtures, with each mixture containing 200 synthetic feature vectors. We assumed the vectors coming from a same mixture belong to the same type. Figures 3(a)-3(c) show the distributions of the 3 set of synthetic feature vectors. From the figures, we can see that the global distribution of the test set 1 has an elliptical shape where the local distributions of all mixtures are spherical, the set 2 has an elliptical global distribution where the local distributions of all mixtures also are elliptical with the same major axis direction, and the set 3 has an irregular global distribution where the local distributions of the mixtures are different.

4.2 Estimation of the Test Data

We applied EM algorithm along with the BYY mixture number selection criterion to estimating the distribution of each test set. Figures 3(d)-3(f) show the value of cost function Eqn. (20) for selecting the number of mixtures in test set 1 to set 3. In our experiments, the cost function $J(k)$ reaches its global minimum at $k = 6$ in the $J(k)$ versus k curves for all test set, where k is the mixture number. From the experimental results, the BYY criterion has correctly chosen the actual number of mixtures for all of our synthetic data sets. Based on the optimal selected mixture number, we estimated the parameters of the Gaussian mixture models corresponding to each test data set.

4.3 Performance Test of the Gaussian Mixture Distance

Under the Gaussian mixture model of a test data set, we can formulate a Gaussian mixture distance function corresponding to a query according to Eqn. (16). Then the Gaussian mixture distance can be used to find out the predefined number of similar vectors from the test data set with the nearest-neighbor search technique. In our experiments, we compared the performance of the Gaussian mixture distance with the Euclidean and Mahalanobis distance for IR. We used the *Precision* of IR as the performance measure which is defined as:

$$\text{Precision} = \frac{\text{Number of target data retrieved}}{\text{Number of data retrieved}}. \quad (21)$$

For each test data set, we conducted 600 trials divided into 5 group to perform nearest-neighbor search using Euclidean, Mahalanobis, and Gaussian mixture distance, where each group contains 120 randomly chosen vectors from the the test set as queries. In each trial, we retrieved 200 similar vectors for a query vectors. Table 1 presents the average *Precision* of the IR results and the average KL divergences of Eqn. (12) using the three distance functions.

From the experimental results, we can see how the local distributions influence the accuracies of the distance functions. The Euclidean distance obtained the higher *Precision* than the Mahalanobis distance for the test set 1, where the local distributions are all spherical. And the Mahalanobis distance has higher *Precision* than the Euclidean distance for the test set 2, where the local distributions are similar with the globe one's. For the test set 3, the *Precision* measurements for the Euclidean and Mahalanobis distance are low since the local distributions are irregular. However, the Gaussian mixture distance always can obtain the highest *Precision* in all the 3 test sets. Our experiments are conducted using an Ultra 5/270 with the MATLAB Version 5.2.1 software. The average time for performing each trial is 0.4591 seconds for the Euclidean, 0.4623 for the Mahalanobis and 0.4628 for the Gaussian mixture distance.

Next, we demonstrate that the Gaussian mixture distance has the variable parameters. Unlike Euclidean or Mahalanobis distance function, the parameters of the Gaussian mixture distance can change along with the position of current query in the feature space. The variable parameters are able to capture the characters of the local distributions of the data. Figures 4(a)-4(c) display the different Gaussian mixture distance contours corresponding to 3 queries at different position in the feature space.

Set No.	Distance	Precision	KL Divergence
1	Euclidean	0.8246	43.9738
	Mahalanobis	0.7210	50.1851
	Gaussian Mixture	0.8279	35.7958
2	Euclidean	0.7278	30.9869
	Mahalanobis	0.7837	22.6229
	Gaussian Mixture	0.7947	13.0392
3	Euclidean	0.6437	34.0311
	Mahalanobis	0.6750	31.3070
	Gaussian Mixture	0.7718	15.1037

Table 1: The average *Precision* of the nearest-neighbor search results using the Euclidean, Mahalanobis and Gaussian Mixture distance for the test data. The average KL divergences of Eqn. (12) are given in the table.

5 Conclusion

In this paper, we formulated a Gaussian mixture distance function based on minimizing the KL divergence between the distribution of the retrieved data and the data in database. Using the Gaussian mixture distance, we can perform more accurate nearest-neighbor search in IR to find out the similar data of a query than using the Euclidean or Mahalanobis distance. Our experiments have shown that the Gaussian mixture distance always has the highest precision performance among the three distance functions for all test data sets. We also noticed that although the Gaussian mixture distance have shown its advantage in accuracy, there still exists an infeasible factor when applying the Gaussian mixture distance to a practical IR application. The weakness is that the computation complexity is high, since we have to calculate the distance between the query and each data in the test data set in our experiments. After all, it is impractical to visit all data of a huge database corresponding to each query. We are now considering to build some optimal indexing structures according the local distribution of data to solve the problem.

References

- [1] I. J. Cox, J. Ghosn and P. N. Yianilos: Feature-based face recognition using mixture-distance. Computer Vision and Pattern Recognition (CVPR), 1996.
- [2] L. Xu: How many clusters?: A YING-YANG machine based theory of a classical open problem in pattern recognition. Proceeding of IEEE International Conference on Neural Networks, 1996.
- [3] L. Xu: Bayesian YING-YANG machine, clustering and number of clusters. Pattern Recognition Letters, 18, 1167-1178, 1997.

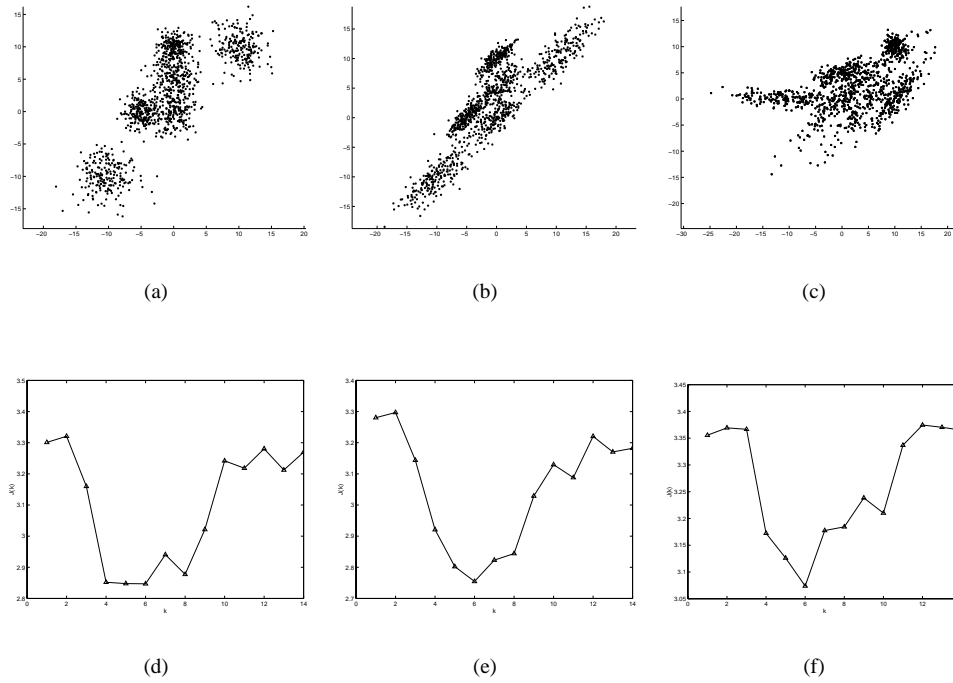


Figure 3: Three test sets of 2-dimensional synthetic feature vectors: (a) the set 1, (b) set 2 and (c) set 3 with the relevant cost function curves (d) corresponding to (a), (e) to (b) and (f) to (c).

[4] S. Kullback: Information theory and statistics. Wiley publication in mathematical statistics, New York, (1959).

[5] A. P. Dempster, N. M. Laird and D. B. Rubin, Journal of the Royal Statistical Society, Series B 39, 1-38, 1977.

[6] H. Akaike: A new look at the statistical model identification. IEEE Transactions on Automatic Control AC-19, 716-723, 1974.

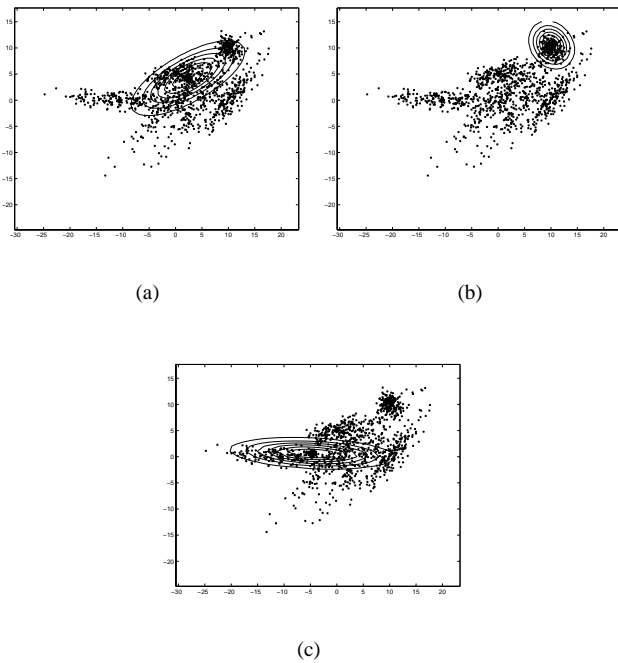


Figure 4: The different shapes of distance contour for the three queries at different positions in the feature space.