Robust Nonparametric Relevance Feedback for Image Retrieval

Ashwin T.V., S. Ghosal, A. Sarkar — IBM India Research Lab, Indian Institute of Technology, Hauz Khas, New Delhi, INDIA

N. Jain* — Dept. of Comp. Sc. and Engr., Indian Institute of Technology, Hauz Khas, New Delhi, INDIA

S. Sarkar * — Dept. of Comp. Sc. and Engr., Indian Institute of Technology, Powai, Mumbai, INDIA

Abstract

Most interactive, "query-by-example" based image retrieval systems employ relevance feedback technique for bridging the gap between the user-defined high-level concept and the low-level image representation in the feature space. We propose in this paper a unified relevance feedback methodology that offers flexibility in capturing user perception and at the same time robustness to deal with limited training images. A generalized additive model based nonparametric probabilistic approach is adopted for flexibility. A generalized ellipsoid based parametric model with outlier rejection is proposed for robustness. Our approach initially assumes a unimodal user perception, and depending on the size of the outliers infers a multi-modal perception, and switches to nonparametric mode. Experimental results with simulated training set are presented to demonstrate the validity and effectiveness of the proposed relevance feedback technique. We also report results on real image databases and show the effect of our algorithm on the end-to-end retrieval performance.

1. Introduction

Rapid growth in the number and size of image databases has created the need for more efficient search and retrieval techniques, since conventional database search based on textual queries can at best provide a partial solution to the problem. This is because either the images are often not annotated with textual descriptions, or the vocabulary needed to describe a user's implied "concept" may not exist (or, at least not be known to the user). Additionally there is rarely a unique description that can be associated with an image.

The standard paradigm for content-based image retrieval (CBIR) is the so-called "query by example" where the user provides the system with one or more query images, and the system retrieves from the database images that are visually similar to the example(s). Initial designs of CBIR engines concentrated largely on selection of feature space so that images that are "close" to each other in feature space are also visually closer. Over the last couple of years, it has widely been recognized that a fully automatic, "rigid" approach to image retrieval cannot satisfy the information need of a wide variety of users. Thus the human-in-the-loop interactive approach has emerged as a *de facto* standard methodology in more recent CBIR engines to bridge the gap between the user-level high-level concept and the low-level representation of images in the feature space. Relevance feedback, commonly practiced in the information retrieval (IR) community has been adopted by the CBIR community as the means of user interaction – given a user's preferences (likes/dislikes) to a set of images, "similar" to his query image (according to an initial metric), the goal is to learn her notion of similarity, and improve the relevance of the retrieved images to that user over successive iterations.

Commonly used relevance feedback techniques can be broadly categorized into two: (1) geometric similarity-based, and (2) probabilistic similarity-based. In geometric similarity-based approaches, a parametric distance measure is assumed between two image feature vectors X_1 and X_2 such as

$$d(X_1, X_2) = (X_1 - X_2)A(X_1 - X_2)^T \qquad (1)$$

where A is the cross-correlation matrix. A is diagonal if feature independence is assumed. In a typical retrieval scenario, given a query feature vector q, a distance $d(X, q)$ is computed between q and an image feature vector X in the database. K images with least distances are retrieved, and user's opinion is sought (A rather interesting query modification-based retrieval paradigm has been recently reported in [5]). The matrix entries (denoting feature weights) are updated based on some feature properties of the relevant images. In MARS [3], one of the earliest CBIR systems employing relevance feedback the feature weights are chosen inversely proportional to the variance of the feature values in the relevant set, based on the intuitive argument that if the relevant images are distributed widely along a feature dimension, its importance in user's mind must be

*Authors performed the work as Summer interns at the IBM India Research Lab.

small. MindReader formalized the MARS approach, and obtained the weight updation rule for non-diagonal A by solving a minimization problem [4]. Rui *et. al* introduced the concept of feature classes, and inter- and intra-feature weights, and showed its effectiveness of the approach over MindReader for image retrieval in [2]. Most of these geometric similarity-based approaches only handle unimodal queries, and ignore the information from the non-relevant set of images. Using information contained in the non-relevant examples is essential to prevent the algorithm from getting stuck in local minima. This also ensures that the algorithm explores the feature space. For example, say an user is interested only in images containing sun setting over the sea and not in images of sunsets over land. This can only be handled by algorithms which explicitly consider the non-relevant images.

The probabilistic approaches on the other hand utilizes the notion of likelihood-based similarity measures. Given a user's preferences on an initial set of images, the probability density functions for relevant and non-relevant images are obtained. Next the likelihood of two image feature vectors being similar or dissimilar – one being the query feature vector q and the other one being an image feature vector x in the database is computed using these distributions. Most likely relevant images are shown in successive iterations. Nastar *et. al* proposed a forward model of density estimation in [8], and integrated both the positive and negative examples in a common parametric density estimation technique. This technique was subsequently enhanced by using nonparametric density estimation in [7], and Bayesian inference was performed to retrieve most probable relevant images from the database. In general parametric methods provide robustness over nonparametric ones, but at the expense of flexibility.

We propose a novel approach to relevance feedback in this paper that offers advantages over existing methods in terms of both robustness and flexibility by integrating the geometric and probabilistic similarity measures. Initially a unimodal perception is assumed and a novel geometric similarity-based search is invoked that explicitly utilizes the information from the non-relevant images. Specifically, relevant images that are close to non-relevant images than others to its class are removed as outliers and only rest of the relevant images are used for updating the entries of cross-correlation matrix A. A nonparametric probabilistic method is invoked if the ratio of outliers and relevant images crosses a threshold. A generalized additive model-based approach is adopted for retrieving the most probable relevant images. The individual performances of both the proposed approaches are established over several existing ones. Performance measurement is conducted using simulated datasets (to decouple the feature selection process from end-to-end retrieval performance) as well as real images.

2 Outlier Rejection Based Parametric Relevance Feedback Technique

We propose a novel geometric similarity-based relevance feedback technique in this Section that explicitly uses the information about non-relevant image feature points. The proposed approach iteratively updates the parameters of a geometric similarity metric so as to fit the relevant feature vectors while excluding the non-relevant ones. This is achieved by modifying the weights associated with the relevant examples. Specifically, relevant points that are far away from the non-relevant points are given more weights, and relevant points that are close to non-relevant ones are treated as outliers and given small weights.

Let $X_1, X_2, \ldots, X_n \in \mathcal{R}^d$ denote a set of image feature vectors, and $Y_1, Y_2, \ldots, Y_n$ be the corresponding user responses. d is the dimension of the feature space. $Y_j = 1$ if X_j is relevant to the user. $Y_j = -1$ if X_j is non-relevant to the user. Further assume X^r and X^n represent the set of relevant and non-relevant feature vectors. $w^r \in \mathcal{R}^{|X^r|}$ represent the weights associated with the relevant examples. Initially $w^r = 1$ for all relevant feature vectors. We assume a generalized ellipsoid-based similarity measure of the form, given by Equation 1. Then, given a feature vector X in the database, its relevance can be computed as

$$d(X; \mu, A) = (X - \mu)^T A (X - \mu) \qquad (2)$$

where $\mu \in \mathcal{R}^d$ represents the target concept, and entries of A represent the feature weights that capture the user perception. Our goal is to estimate μ and A, so that the sum of distances of relevant vectors from the target concept is minimized. Or in other words, minimize

$$\mathcal{J} = \sum_{i=1}^{n_g} w_i^r (X - q)^T A (X - q)$$

w.r.t. q and A, subject to the constraint $\det(A) = 1$. It can be shown [4],

$$\mu(X^r, w^r) = \frac{\sum_{x_i \in X^r} w_i^r x_i}{\sum w_i^r}. \qquad (3)$$

and

$$A(X^r, w^r) = C^{-1}. \qquad (4)$$

where, C is the covariance matrix for the set X^r of relevant images, given by

$$C(X^r, w^r) = \frac{\sum_{j=1}^{n_g} w_j (X_j - \mu)(X_j - \mu)^T}{\sum_{j=1}^{n_g} w_j^r}. \qquad (5)$$

In the above formulation, w_j^r's are provided by the user in each iteration. Typically existing systems require the user to rank images based on their relevant to her, from very relevant to not so relevant. We feel this multi-level relevance assignment is too difficult for a user and prone to error. Recently Hong *et. al* proposed a Support Vector Machine (SVM) based technique for automatic weight assignment for relevant images [1], where the output of the trained SVM for each relevant example is used as its weight. The output of the SVM classifier gives the distance of the input feature vector from the separating hyperplane in a transformed domain. Weighing the examples using this distance to estimate a quadratic metric may not be meaningful in the original space. Also, the authors do not address the issues associated with using a small training set obtained from the user's feedback in training a SVM classifier.

2.1 Automatic weight updation for relevant images

We propose a new technique where only binary relevance assignment is sought from the user, and the weights w^r are automatically computed based on their distances from the non-relevant images. This proposed method updates the weights (w^r) and the parameters of the similarity metric iteratively, so that the ellipsoids represented by the successive similarity metrics better capture the positive examples while excluding negative ones. The algorithm, as shown in Fig. 1 begins by initializing all the weights w_i^r to one, i.e. initially all the relevant examples are considered to be equally important. In each iteration, the parameters of the similarity metric (μ and A) using the current weight vector w^r are determined. The distances of the relevant and the non-relevant examples from the learned target concept (μ) are determined using (2). Let x^r_{max} denote the farthest positive example having a *non-zero weight* and d^r_{max} be its distance from μ. Let $\mathcal{E}$ be the ellipsoid defined by (μ, A) and having a radius d^r_{max}. Let X^n_{new} represent the set of negative examples which fall inside the ellipsoid $\mathcal{E}$. The aim of the algorithm is to modify the parameters (μ, A) in each iteration to reduce the number of such examples. This is achieved as follows. The weight of the farthest positive example x^r_{max} is set to 0. The weights of the other positive examples with non-zero weights are updated as the sum of their quadratic distances from the examples in X^n_{new}. The updated weights are then used to obtain a new estimate of the similarity metric parameters and the iteration proceeds. The iteration stops when the size of X^n_{new} becomes zero.

The algorithm proceeds by removing a positive example in each iteration. This positive example is considered an "outlier" since its inclusion in the estimation of the similarity metric results in negative examples (X^n_{new}) having smaller distances than positive examples. This would lead to the examples in X^n_{new} being retrieved again in the next iteration. To avoid this, the remaining relevant examples are weighted by their cumulative distances from examples in X^n_{new}. Hence, the metric estimated in the next iteration is forced away from X^n_{new}.

```
Input:   X^r, X^n the relevant and non-relevant examples.
Output: A and μ parameters of similarity metric.
Let min_d be the dimension of the feature space
    class with smallest number of components;
Let w^r ∈ R^{|X^r|};  w^r = 1;
while (1) {
        Calculate A(X^r, w^r) using (4);
        Calculate μ(X^r, w^r) using (3);
        X^r_max = argmax_{X ∈ X^r} {d(X; μ, A)}
                  where d given by (2)
        d^r_max = d(X^r_max; μ, A);
        X^n_new = {X : X ∈ X^n; d(X; μ, A) < d^r_max};
        if ( |X^n_new| = 0 ) break;
        w^r[x^r_max] = 0;
        w^r_i = { 0                               if(w^r_i = 0)
                { Σ_{Z ∈ X^n_new} d(X^r_i, Z)     otherwise
                  d(X, Z) = (X − Z)^T A(X − Y)
        if ( |{w^r_i : w^r_i != 0}| < min_d )
                  break;
}
return A and μ.
```

Figure 1: Proposed weight updation algorithm.

The algorithm stops when either of the following conditions are satisfied:

1. There exist no negative examples inside $\mathcal{E}$, i.e. X^n_{new} is empty. We have achieved our objective of determining the parameters (μ, A) to best fit the set of relevant examples and excluding the irrelevant examples.

2. The number of *non-zero weighted* positive examples is so small that some of the matrices C_i in (4) become singular. This happens when the number of positive examples are too small or when they are distributed in the image space.

The algorithm is greedy in nature since the *farthest* positive example is removed in every iteration. Other methods to search for the best subset of relevant examples can also be employed. Jolion [6] describe a random sampling based approach.

Refer to Fig. 2 to see how the proposed approach selects outliers and iteratively generates the similarity metric. The dataset consists of 24 relevant and 25 non-relevant examples. Based on their proximity to non-relevant points, 13 relevant images were declared to be outliers (marked as dark squares), and a similarity metric is derived. This is depicted as an ellipse in the Figure. Evidently in the first

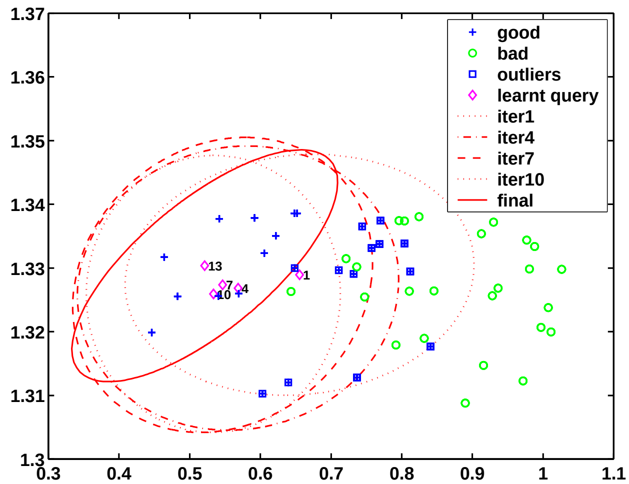

Figure 2: Iterative learning of similarity metric on a synthetic 2-d dataset

iteration 11 non-relevant examples are enclosed, which reduces to zero after 13 iterations.

3 Generalized Additive Model-based Non-Parametric Relevance Feedback

A novel probabilistic relevant feedback method is proposed in this Section that utilizes generalized additive models for learning user's perception given the user's preference on a set of images. This proposed technique is inherently non-parametric in nature, that makes little assumptions about the underlying distribution. Thus this approach is natural candidate for handling multi-modal user queries, in which user's preference cannot be approximated using an ellipsoid in feature space.

The probabilistic approach estimates the likelihood of an image to be relevant to the user, given a user's responses to a set of images. We assume a binary response from the user - a 1 for relevant images, and a 0 for the non-relevant ones. Thus we define a binary random variable Y, which would take the value 1 at a feature point corresponding to a relevant image and the value 0 at a non-relevant feature point. Or, in other words, the probability of an image feature vector X to be relevant is $P(X) = Prob(Y = 1 \mid X)$. Similarly the probability of an image feature vector X to be marked non-relevant is $Prob(Y = 0 \mid X)$. Obviously,

$$Prob(Y = 1 \mid X) + Prob(Y = 0 \mid X) = 1$$

Let $X_1, X_2, \ldots, X_n$ be a set of image feature vectors, and $Y_1, Y_2, \ldots, Y_n$ be the corresponding user responses (1 for relevance, 0 for non-relevance).Then the likelihood L of the current estimates for $P(X_i)$ is given by the joint probability of $Y_1, Y_2, \ldots, Y_n$

$$\mathcal{L} = \prod_{i=1}^{n} P(X_i)^{Y_i} \left(1 - P(X_i)\right)^{(1-Y_i)},$$

or, more conveniently,

$$L = \sum_{i=1}^{n} Y_i \log(P_i) + (1 - Y_i)\log(1 - P_i), \quad (6)$$

where $P_i = P(X_i)$. The goal here is to estimate the function $\hat{P}(.)$ that maximizes the log-likelihood L in 6. Using this maximum likelihood (ML) estimate $\hat{P}(.)$, relevance of entire database feature vectors are estimated, and the maximally relevant images are retrieved.

3.1 Non-Parametric ML Estimation of P

Most of the reported work in the literature assume some parametric form for the probability function, the sake of convenience being one of the main reasons. Parametric models also offer robustness in the context of interactive image retrieval, given the lack of training samples (a user cannot be expected to mark too many images as relevant or non-relevant during a session). However, it is not likely that information need of a diverse set of users can be satisfied using parametric models (even if a set of choices is given to the user, it is not practical to assume that the user can select one). For example, if a multivariate Gaussian form for P is assumed [7], retrieval performance will be satisfying a user, only when the user perception can be expressed as a Gaussian cluster in the feature space. Similarly if a generalized linear form for P is assumed, only a simple hyperplane in the feature space can be realized. In this paper, we propose a generalized additive model (GAM) based flexible method for learning user's perception.

Before getting into details, let us introduce the concept of a logistic function from the statistics literature. Our problem is to fit a function $P = F(X)$. By definition, the function $0 \le F(.) \le 1$. To simplify the form of the function, define a logistic function $\eta(X)$ such that

$$\hat{P}(X) = \frac{e^{\eta(X)}}{1 + e^{\eta(X)}}. \quad (7)$$

Note that $-\infty \le \eta(.) \le \infty$. The inverse is given by the *logit* function,

$$\eta(X) = \log\left(\frac{\hat{P}(X)}{1 - \hat{P}(X)}\right). \quad (8)$$

This function has the properties of monotonicity and being one–to–one and onto. It also has a simple intuitive meaning

Set $\hat{g}_j^{(0)} \leftarrow$ zero function
$\hat{\alpha}^{(0)} \leftarrow logit(\bar{Y}), \quad \bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$
Repeat over loop variable k
for $i = 1$ to n
$\eta^{(k)} = \hat{\alpha}^{(k)} + \sum_{j=1}^{d} \hat{g}_j^{(k)} X_{j,i}$
$P_i^{(k)} = logit^{-1}\left(\eta^{(k)}\right)$
$w_i = P_i^{(k)}$
$Z_i = \eta^{(k)} + \left(Y_i - P_i^{(k)}\right)/w_i$
Set $\hat{g}_j^{(k+1)}, \hat{\alpha}^{(k+1)}$ by backfitting Z_i on X_i
with weights w_i
converges

Figure 3: Local Scoring algorithm

$\hat{g}_j(x) = 0, \forall j$
$\hat{\alpha} = \frac{1}{n}\sum_{i=1}^{n} Y_i$
REPEAT
FOR $j = 1$ TO d
FOR $i = 1$ TO n
$\hat{\epsilon}_i = Y_i - \hat{\alpha} - \sum_{k \neq j}^{d} \hat{g}_k(x_{k,i})$
END
$\hat{g}_j = \text{SMOOTH}(x_{j,\cdot}, \hat{\epsilon}_i)$ [10](pages 41-42)
END
UNTIL $\mathcal{E} = \sum_i (Y_i - G(X_i))^2 < \tau$

Figure 4: Back Fitting Algorithm

as the logarithm of the odds of $\hat{P}$. The sigmoid form of the function, takes value 1 for $x > 0$ and 0 for $x < 0$ and is still a continuous function (analogous to neural network classification model). Now, the log-likelihood functional L, given in Equation 6 can be rewritten as

$$L = \sum_{i=1}^{n} Y_i \eta(X_i) - log(1 + e^{\eta(X_i)}) \qquad (9)$$

3.1.1 Generalized Additive Model

In the Generalized Additive Model (GAM) of Hansie and Tibshirani [9], $\eta(X)$ is expressed as a sum of general functions $g_j(.)$ in each feature dimension X_j, i.e.,

$$\eta(X) = \alpha + \sum_{j=1}^{d} g_j(X_j), \qquad (10)$$

where, d is the number of feature dimensions, and α is a constant to be determined. Thus, the goal now is to solve for α and family of functions $g_j(.)$ so that the following log-likelihood functional is maximized

$$\begin{aligned} lclL &= \sum_{i=1}^{n} Y_i(\alpha + \sum_{j=1}^{d} g_j(X_{j,i})) \\ &- \log(1 + \exp(\alpha + \sum_{j=1}^{d} g_j(X_{j,i}))). \end{aligned} \qquad (11)$$

Hastie and Tibshirani's local scoring algorithm is used here to efficiently solve the maximization problem (Equation 11). This local scoring algorithm is shown in Figure 3.1.1. An iterative backfitting algorithm [10], shown in Fig. 3.1.1 is chosen in this paper. During each iteration the basic idea is to fit one dimension at a time, estimate how much the contribution of the current dimension should be, fitting a curve in that dimension, and repeat this for all dimensions. This is repeated till the error falls below a preset threshold τ. The local scoring algorithm then uses the backfitting algorithm to fit values proportional to the values that are left to be fitted, that is, points where the current model does not perform well are given higher weights w_is.

The family of functions g_j still remain to be estimated. One of the ways to generate a function is to fit kernel smoothers over the data. These are a sort of moving–average functions that consider the function to be a average of nearby points. Gaussian kernel smoothers have been used in [7]. One problem with Gaussian smoother is that if training points are sufficiently far apart, the estimate of the function would be wrong, as the Gaussians would go to zero. Instead, spline smoothers are chosen to be the smoothing function in this study. Splines have been shown to have the form of a kernel smoother. Further, they can be evaluated at new points quite easily. They are also easy to calculate, taking time linear in the number of training points n [11].

Given the values Y_i at the data points X_i, without loss of generality, assume $a < X_1 < X_2 < \ldots < X_n < b$. The natural cubic spline smoother is defined to be the function minimizing the expression

$$\mathcal{J}(g_j) = \sum_{i=1}^{n} w_i (Y_i - g_j(X_{j,i}))^2 + \alpha \int_a^b g_j^{''}(t)^2 dt, \quad (12)$$

where w_i is the weight associated with the i-th point, and α is a parameter that determines the amount of 'smoothing'. A very low value would give a good fit to the data, while not doing well for points outside the training sample. A very high value would on the other hand be almost linear, missing the variations in the data (see Figure 8.

It is well known that the solution is the function $\hat{g}$ that is cubic in each of the intervals $[X_1, X_2], [X_2, X_3], \ldots, [X_{n-1}, X_n]$ with the additional constraints that its first and second derivatives are all continuous at each of the X_i, and second and third derivatives are zero at the end points X_1 and X_n.

The spline function can easily be specified by the values of the function and its second derivatives at each of the points X_i, which are called “knot points”, i.e. the vectors $\mathbf{G}$ and γ, where

$$\mathbf{G}(i) = \hat{g}_j(X_i), \text{and}$$
$$\gamma(i) = \hat{g}_j^{''}(X_i).$$

This form allows us to use the Reinsch algorithm (see [10],pages 41-42) for a linear time solution to the problem.

As mentioned earlier, nonparametric methods are inherently not robust in the sense that the estimate are not reliable at points far away from training samples. Thus if an image feature vector is far away from all the feature vectors, marked as the user, the spline smoothing may give poor results. We thus try to constrain the system so far distant feature points do not arbitrarily achieve high probability of being selected (that will remove the points, currently marked relevant to be excluded in the list of k retrieved images in the next iteration). To alleviate this undesirable behavior of splines, a discontinuity in the spline at the end points are introduced, and by drawing a curve from that point to a negative spline value (negative values give a probability zero after the sigmoid is applied on them). This curve could be simply taken to be a straight line to a boundary, or an exponential cut-off.

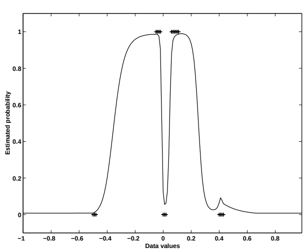

Figure 5: $\alpha = 1$

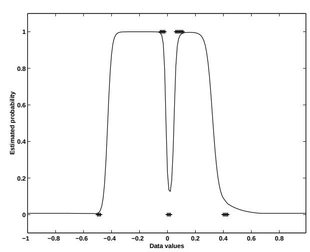

Figure 6: $\alpha = 10$

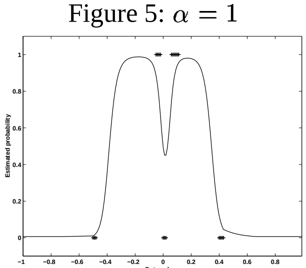

Figure 7: $\alpha = 100$

Figure 8: Spline fitting with different α

4 Combining the Parametric and Non-Parametric Approaches

Both parametric and nonparametric approaches have their own strengths and limitations in the context of image search and retrieval. Parametric methods by their very nature are robust, and work well when there are not enough available training data. This is particularly true in interactive image retrieval framework where the user is not expected to mark too many images based on which the system can learn his intended “concept”. Parametric methods, however, assume an underlying model (in our case generalized ellipsoid), and if this model does not adequately capture the user perception, the retrieval performance becomes clear. On the other hand, nonparametric (generalized additive models) methods offer flexibility in capturing the user perception in an efficient manner but only at the cost of robustness. We propose a new integrated scheme that combines our parametric and nonparametric the two approaches in a complimentary way.

In this proposed integrated relevance feedback framework, the outlier rejection-based estimation algorithm is invoked first, given the user’s responses on the initial set of retrieved images (K Nearest neighbors of the query vector in Euclidean space). This algorithm, according to Fig. 1 computes the weights w_j^r in the first relevance feedback iteration. Now a decision is made based on the distribution of w^j, whether a generalized ellipsoid is sufficient to capture user perception or not. Efron [12] have proposed that to robustly fit a model requiring p parameter estimates, at least $2p$ observations are needed. Hence to obtain the generalized ellipsoid model, we need at least twice the number of the ellipsoid parameters (the elements of the covariance matrix and the mean vector). If after removal of outliers the number of remaining points is less than this threshold, GAM-based nonparametric method is executed. This sequence of steps are repeated, till the user is satisfied with the retrieval result.

5 Experiments

In this section we demonstrate the performance of our algorithms on synthetic and real image datasets.

In the first experiment we present results on artificially generated datasets. Two synthetic datasets were generated, corresponding to two assumed models of users perception. In the first dataset the users perception is modeled as a Gaussian distribution. This models the case where the user’s perception forms a single cluster in the feature space. The second dataset was obtained by assuming a mixture of gaussians model. This models the case where the users perception of relevance is distributed in disjoint clusters in the feature space. The examples in the dataset for both these cases were 2-dimensional and were generated using an assumed beta distribution. The examples in the dataset were then ranked based on the assumed user perception model. Examples having a confidence greater than a threshold were labelled as relevant.

The second experiment uses the Columbia Object Image Library dataset. It contains pictures of 20 objects, each viewed from different positions, giving a total of 1440 images. There are 72 views per object. The images are of size 128×128. The images after 8×8 block averaging are represented as vectors in a 256 dimensional feature space. The feature space dimension is then reduced to 4 through PCA. In a particular experiment one of the objects was labelled as relevant. The objective is to retrieve the views corresponding to this object with a small number of relevance feedback iterations. The experiment was repeated for different objects.

The relevance feedback loop in the two experiments works as follows. The starting queries in the experiments were distributed along the boundary of the relevant examples. In the first iteration of the relevance feedback loop, examples ranked based on their Euclidean distances from the starting query are retrieved. These examples are labelled and input to the learning algorithm. Examples in the dataset are then ranked by the learning algorithm. A fixed number of top ranked examples are retrieved. After labelling, the retrieved examples are input to the learning algorithm, closing the relevance feedback loop.

We compare the performance of the learning algorithms we have proposed in this paper with algorithms proposed by Rui et. al. [3], Meilhac and Nastar [8] and by Ishikawa et. al. [4].

5.1 Simulated dataset results

Fig.(9) plots the artificially generated 2d dataset. The coordinates of the points along each axis conform to a mixture of 2 beta distributions.

5.1.1 Unimodal model

A Gaussian distribution centered at $(0.3, 0.3)$ was assumed to model relevant examples. The ellipse obtained by thresholding the Gaussian is shown in Fig.(10). The points inside the ellipse are labelled relevant. Fig.(10) also plots the starting query points. The relevance feedback loop was simulated starting from each of these points. The learning algorithm in each iteration is input 50 labelled examples. To determine the generalization performance, the number of relevant examples among the 150 highest ranked examples is used. The number of relevant retrieved examples in an iteration of the relevance feedback loop was averaged over different starting points. Precision is defined as percentage of retrieved examples which are relevant. Precision over successive iterations is plotted for different algorithms in Fig.(12).

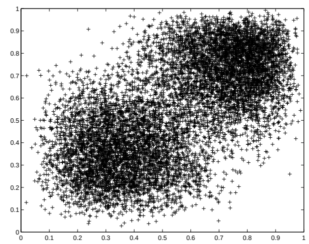

Figure 9: Data points in synthetic dataset

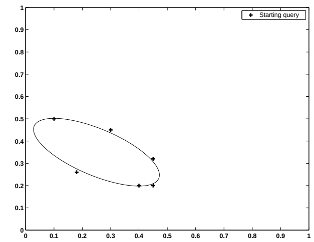

Figure 10: Unimodal model of User's perception.

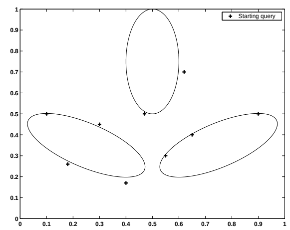

Figure 11: Multimodal model of User's perception.

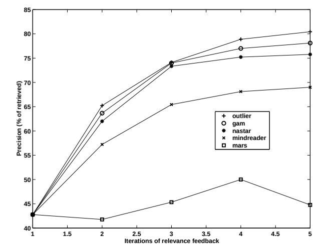

Figure 12: Precision over relevance feedback iterations for Unimodal model.

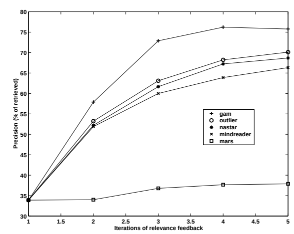

Figure 13: Precision over relevance feedback iterations for Multimodal model.

5.1.2 Multimodal model

Three Gaussian distributions centered at $(0.5, 0.75), (0.25, 0.35), (0.75, 0.35)$ were used to model relevant examples. The corresponding relevant regions are represented by the ellipses shown in Fig.(13). The learning algorithm is trained on 50 examples and tested on 150 examples. The precision performance over successive iterations is shown in Fig.(13). The nonparametric GAM algoritm shows the best performance in this case.

5.2 COIL dataset

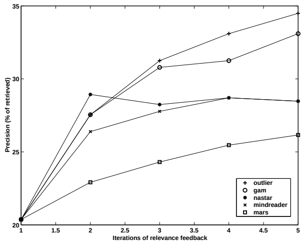

Figure 14: Object number 2

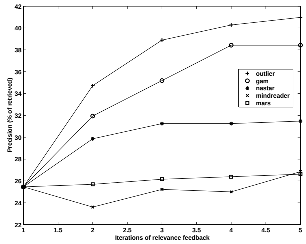

Figure 15: Object number 3

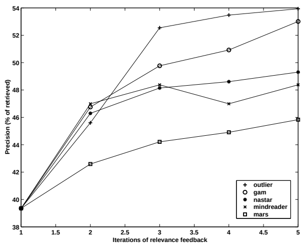

Figure 16: Object number 4

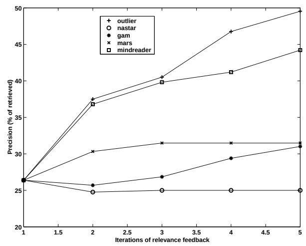

Figure 17: Object number 6

Figure 18: Retrieval of views of objects from COIL dataset

Experiments to retrieve the views of different objects were performed. The retrieval results are shown in Fig.(14), Fig.(15),Fig.(16),Fig.(17).

6 Discussions

We have proposed a unified relevance feedback technique that offers flexibility in capturing user perception without sacrificing robustness too much. This proposed approach switches between parametric and nonparametric based on an outlier-based measure. Our method starts in a generalized ellipsoid-based parametric mode, given a query image. This parametric method uses information from the non-relevant images to automatically weigh the relevant points. Relevant points that are closer to the non-relevant points than others in its set are treated as outliers during adjustment of the similarity metric. If the ratio of outliers and the relevant points reaches a high value, then our method infers a multi-modal user perception, and switches to a generalized additive model-based nonparametric mode. Experiments with simulated as well as real image data show the validity and relative performance of the proposed approach.

There has been a lot of progress in computer vision and pattern recognition communities in the area of classification, face detection being a more recent example of that. Note that while fundamentally similar in nature, interactive "query-by-example"-based image retrieval paradigm poses new challenges, since the training size is very limited, and moreover the training points are not necessarily representative of the database feature vectors. Thus incremental classification with few training samples and good generalization become the key issue. All existing relevance feedback techniques, including ours choose a greedy approach, and try to efficiently classify the training samples at each iteration. Our current research focus include application of active learning principles and that would reduce user interaction by seeking feedback on an intelligently selected more *informative* set of images, and principles of learning from incomplete data that would reduce the number of feedback iteration.

References

[1] P. Hong and Q. Tian and T. Huang, "Incorporate Support Vector Machines to Content-Based Image Retrieval with Relevance Feedback," *Proc. IEEE ICIP*, 2000.

[2] Y. Rui and T. Huang, "Optimizing Learning in Image Retrieval," *Proc. IEEE CVPR*, 2000.

[3] Y. Rui and T.S. Huang and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS," *Proc. IEEE ICIP*, 1997.

[4] Y. Ishikawa and R. Subramanya and C. Faloustos, "Mindreader: Query databases through multiple examples," *Proc. 24th VLDB conference*, 1998.

[5] G. Aggarwal and P. Dubey and S. Ghosal and A. Kulshreshta and A. Sarkar, "iPURE: Perceptual and User-friendly REtrieval of Images," *Proc. IEEE Int. Conf. on Multimedia and Exposition*, 2000.

[6] J. M. Jolion and P. Meer and S. Bataouche, "Robust Clustering with applications in computer vision," *IEEE T PAMI*, Vol. 13, NO. 8, 1991.

[7] C. Meilhac and C. Nastar, "Relevance Feedback and Category Search in Image Databases," *IEEE Int. Conf. on Multimedia Computing and Systems*, 1999.

[8] C. Nastar and M. Mitschke and C. Meilhac, "Efficient Query Refinement for Image Retrieval," *Proc. IEEE CVPR* , 1998.

[9] Trevor Hastie & Robert Tibshirani, *Generalised Additive Models*.

[10] W. Hardle, *Applied Nonparametric Regression*, Cambridge University Press,1990.

[11] P. J. Green & B. W. Silverman, *Nonparametric Regression and Generalised Linear Models*, Chapman and Hall, 1994.

[12] B. Efron, "Double exponential families and their use in generalized linear regression", *Journal of the American Statistical Association, Vol. 81*,1986.