Learning Similarity Measure for Natural Image Retrieval with Relevance Feedback

Guo-Dong Guo†, Anil K. Jain‡,
† Micorsoft Research China
5F, Beijing Sigma Center, P. R. China

Wei-Ying Ma†, Hong-Jiang Zhang†
‡ Department of Computer Science & Engineering
Michigan State University

Abstract

A new scheme of learning similarity measure is proposed for content-based image retrieval (CBIR). It learns a boundary that separates the images in the database into two parts. Images on the positive side of the boundary are ranked by their Euclidean distances to the query. The scheme is called restricted similarity measure (RSM), which not only takes into consideration the perceptual similarity between images, but also significantly improves the retrieval performance based on the Euclidean distance measure. Two techniques, support vector machine and AdaBoost, are utilized to learn the boundary, and compared with respect to their performance in boundary learning. The positive and negative examples used to learn the boundary are provided by the user with relevance feedback. The RSM metric is evaluated on a large database of 10,009 natural images with an accurate ground truth. Experimental results demonstrate the usefulness and effectiveness of the proposed similarity measure for image retrieval.

1. Introduction

Content-based image retrieval (CBIR) has been an active research issue in computer vision [11] [13] [5] [7] [18]. In retrieval, there is typically a user in the loop. The image retrieval system should therefore take into consideration human perceptual similarity between the query and the retrieved images. Thus the retrieval process is subjective in a sense [5]. Relevance feedback (RF) is a powerful technique for interactive image retrieval [16]. Minka and Picard [10] presented a learning technique for interactive image retrieval. The key idea behind this approach is that each feature model has its own strength in representing a certain aspect of image content, and thus, the best way for effective content-based retrieval is to utilize "a society of models". A typical approach in relevance feedback is to adjust the weights of various features to accommodate the user's need [15] [16]. Another method is to modify and convert the query into a new representation by using the positive and negative examples provided by the users [15]. In [4], relevance feedback is used to modify the weighted metric for computing the distance between feature vectors.

In this paper, we propose a technique that learns a boundary to separate the positive and negative examples provided by relevance feedback. Support vector machine and AdaBoost are used to learn the boundary which is utilized to filter the images in the database. Another approach to filtering is to classify the images in the database into semantic or high-level categories [24]. The key idea is to restrict the images used for similarity measure with respect to the query. We first provide our motivation in next Section. Then, we introduce our scheme for image representation in Section 3, and the metric of restricted similarity measure in Section 4. The performance of the proposed method is evaluated in Section 5. Finally, we give the conclusions.

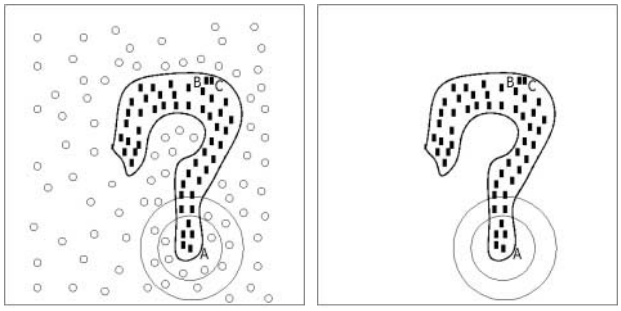
2. Motivation of Our Approach

Similarity measure is a key component in image retrieval. Traditionally, Euclidean distance is used to measure the similarity between the query and the images in the database. The smaller the distance, the more similar the pattern to the query. However, this metric is sensitive to the sample topology, as illustrated in Fig. 1 (a). Assume point "A" is the query, the Euclidean distance based similarity measure can be viewed as drawing a hyper-sphere in the high dimensional feature space (or a circle in 2-D), centered at point "A". The larger the radius of the hyper-sphere, the more images are enclosed in the hyper-sphere, as shown in Fig. 1 (a). The radius is determined indirectly by the number of retrieved images. For different queries, the center of the circles move accordingly. As a result, the retrieved images enclosed by the hyper-sphere are different although these query images are perceptually similar. Furthermore, many irrelevant images could be enclosed by the regular hyper-sphere and presented to the user. To solve these problems, we propose to use an "irregular" non-spherical boundary to separate the similar images from the

ISBN 0-7695-1272-0/01 $10.00 (C) 2001 IEEE

dissimilar ones, and the Euclidean distance measure is applied only to a limited number of images. As shown in Fig. 1 (b), the Euclidean similarity measure for query "A" is only done with respect to the black rectangular patterns.

The boundary can be learned from the positive and negative examples provided by the user in image retrieval. We decide to use learning techniques that are non-parametric and do not need a large number of examples to learn a decision boundary. Large margin classifiers, such as SVM [25] [2] and AdaBoost [3], can be used for such purpose.



(a) (b)

Figure 1. Perceptually similar patterns as rectangular ones. See the text for description.

Can we directly use the distances of the images to the boundary to define the similarity measure? The answer is "no". Suppose a query image "B" is given by the user, which is very similar to image "C", as shown in Figs. 1 (a) and (b). Both "B" and "C" are at the positive side of the boundary, and yet close to the boundary. In such a case, other images with large distances to the boundary will always be ranked in the top n matches when the distance-from-boundary (DFB) metric is used for similarity measure, while image "C" can only be retrieved for example after top 20 matches or even more. In an extreme case, image "C" is the same as "B", but can not be retrieved in the top 1 or 2 matches. On the contrary, if we use Euclidean distance measure for the small number of images filtered by the boundary, the image "C" can usually be retrieved in the top n matches. In other words, the merit of the Euclidean distance measure is lost if merely the DFB metric is used.

3. Image Representation

Color information is one of the important features for image retrieval [21]. We use the HSV color space since it provides the best retrieval performance for color histograms [7]. The color histogram is quantized to 256 levels, which results in 256 features for each image. Color moments constitute another kind of color features, which are simple yet effective for image retrieval [20], and do not require quantization. The first three order moments are calculated in the HSV space of each image, resulting in a feature vector of dimension 9. In addition, color coherence vectors (CCV) is used to incorporate spatial information into color histogram representation [12]. The CCV features with 64 quantization result in a 128-dimensional feature vector.

Texture is another type of low-level image feature. The Tamura features are designed based on the psychological studies in human visual perceptions of textures [22]. We compute the coarseness histogram with 10 quantization levels, and the histogram of directionality with 8 quantization levels. Another one is the wavelet coefficients. The pyramidal wavelet transform (PWT) [9] is used and the mean and standard deviation of the energy distribution are calculated corresponding to each of the sub-bands at each decomposition level. For three-level decomposition, PWT results in a feature vector with 24 $(3 \times 4 \times 2)$ components.

We concatenate all color and texture features into one 435 vector (with normalization) to represent each image.

4. Restricted Similarity Measure

In retrieval, we use the restricted similarity measure (RSM) with the restriction imposed by the boundary between the positive and negative examples.

4.1. Providing Examples

How to provide the system with some positive and negative examples? One way is to present a set of pre-selected positive and negative examples for each query class as in [23]. However, new queries may be on the negative side of the boundary (or even far away) pre-learned with the pre-selected examples, thus the user can not obtain his expected results. A better way is to provide examples using relevance feedback technique, which is natural and adaptive. Therefore the boundary is adapted to each query.

4.2. Learning with Support Vector Machine

Given two-class examples, $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, with $\mathbf{x}_i \in R^d$ and $y_i \in \{-1, +1\}$, support vector machine (SVM) finds an optimal separating hyperplane (OSH) $\mathbf{w}\mathbf{x} + b = 0$ to separate them. The optimal solution is the saddle point of the Lagrange functional, $L(\mathbf{w}, b, \alpha) = \frac{1}{2} \parallel \mathbf{w} \parallel^2 - \sum_{i=1}^{n} \alpha_i \{y_i [(\mathbf{w} \cdot \mathbf{x}_i) + b] - 1\}$, where α_i are the Lagrange multipliers. By Lagrangian duality, the solution

$$\bar{\mathbf{w}} = \sum_{i=1}^{S} \bar{\alpha}_i y_i \mathbf{x}_i, \quad \bar{b} = -\frac{1}{2} \bar{\mathbf{w}} \cdot [\mathbf{x}_p + \mathbf{x}_r] \qquad (1)$$

where $\mathbf{x}_p$ and $\mathbf{x}_r$ are any two support vectors with $\bar{\alpha}_p, \bar{\alpha}_r > 0,\ y_p = 1, y_r = -1$, and $S < n$.

ISBN 0-7695-1272-0/01 $10.00 (C) 2001 IEEE

Slack variables $\xi_i \geq 0$ and a penalty function, $F(\xi) = \sum_{i=1}^{S} \xi_i$, where ξ_is are a measure of the mis-classification error, can be used to solve the non-separable problem [1]. The solution is identical to the separable case except for a modification of the Lagrange multipliers as $0 \leq \alpha_i \leq C,\ i = 1, \ldots, S$. The SVM can realize non-linear discrimination by kernel mapping [25], and we choose the Gaussian radial basis function (GRBF) $K(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{(\mathbf{x}-\mathbf{y})^2}{\sigma^2}\right)$ in our experiments, where σ is the width of the Gaussian.

4.3. Learning the Boundary with AdaBoost

Boosting is a method to combine a collection of weak learners to form a stronger classifier. AdaBoost is an adaptive algorithm, in that the weights are updated dynamically according to the errors in previous learning [3]. Tieu and Viola [23] adapted the AdaBoost algorithm for image retrieval. They let the weak learner work on a single feature each time. So after T rounds of boosting, T features are selected together with the T weak classifiers. The simple algorithm [23] is briefly described as below:

AdaBoost Algorithm

Input: 1) n training examples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ with $y_i = 1$ or 0; 2) the number of iterations T.

Initialize weights $w_{1,i} = \frac{1}{2l}$ or $\frac{1}{2m}$ for $y_i = 1$ or 0, respectively, with $l + m = n$.

Do for $t = 1, \ldots, T$:

1. Train one hypothesis h_j for each feature j with w_t, and error $\epsilon_j = Pr_i^{w_t}[h_j(x_i) \neq y_i]$.

2. Choose $h_t(\cdot) = h_k(\cdot)$ such that $\forall j \neq k, \epsilon_k < \epsilon_j$. Let $\epsilon_t = \epsilon_k$.

3. Update: $w_{t+1,i} = w_{t,i}\beta_t^{e_i}$, where $e_i = 1$ or 0 for example x_i classified correctly or incorrectly respectively, with $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$ and $\alpha_t = \log\frac{1}{\beta_t}$.

4. Normalize the weights so that they are a distribution, $w_{t+1,i} \longleftarrow \frac{w_{t+1,i}}{\sum_{j=1}^{n} w_{t+1,j}}$.

Output the final hypothesis,

$$h_f(x) = \begin{cases} 1 & if \ \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2}\sum_{t=1}^{T} \alpha_t \\ 0 & otherwise \end{cases} \quad (2)$$

However, they [23] did not consider the perceptual similarity between images. In fact, the distance to the decision boundary can not be used directly to measure perceptual similarity as explained in Section 2. Here, we use the AdaBoost [23] to learn a boundary and compare it with the SVM based learning. Furthermore, instead of using preselected images [23] for each class, the boundary is learned with the user interaction.

4.4. Restricted Similarity Measure

For a query, after the boundary is learned based on the user's feedback, the images in the database are filtered by the boundary, and treated differently based on their positions with respect to the boundary. For the positive images, we rank them based on their Euclidean distances to the query. It is well known that in the CIE $L^*a^*b^*$ and $L^*u^*v^*$ color spaces [27], the Euclidean distance between two colors is proportional to their perceptual dissimilarity [14]. Thus the Euclidean distance can be used as a similarity measure for color images. Currently, there are no texture features where the Euclidean distance corresponds to human perceptual dissimilarity, yet intuitively the Euclidean distance can be used for texture similarity measure [8] [6]. On the other hand, the negative images are ranked only based on their distances to the boundary, which comes from the intuition that the images similar to the query may not have positive distances, but typically they are expected not far away from it. So, these images can be retrieved for the user just after the positive images if they are ranked by their distances to the boundary. For this reason, we use the distance-from-boundary (DFB) measure to deal with the negative images. Why do we rank negative images? Two considerations: one is that some perceptually similar images may have negative distances to the boundary. If they are discarded, they may not be retrieved to the user forever; another is that sometimes the user would like to browse more images than the filtered positive images. If the negative images are discarded, the number of images to be retrieved will be insufficient. In sum, our strategy is called restricted similarity measure (RSM). It is formulated as

$$S(\mathbf{x}, q) = \begin{cases} ED(\mathbf{x}, q), & if \ D(\mathbf{x}, \Theta) \geq 0 \\ M - D(\mathbf{x}, \Theta), & otherwise \end{cases} \quad (3)$$

where $S(\mathbf{x}, q)$ denotes the similarity measure of the image $\mathbf{x}$ with respect to the query q, and $D(\mathbf{x}, \Theta)$ represents the distance of $\mathbf{x}$ to the boundary characterized by a parameter set Θ. The distance of the image $\mathbf{x}$ to the boundary $D(\mathbf{x}, \Theta) = 0$ is calculated by

$$D(\mathbf{x}, \Theta) = \sum_{i=1}^{S} \bar{\alpha}_i y_i K(\mathbf{x}_i, \mathbf{x}) + \bar{b} \quad (4)$$

for the SVM learned boundary, and computed by,

$$D(\mathbf{x}, \Theta) = \sum_{t=1}^{T} \alpha_t \left(h_t(x) - 0.5\right) \quad (5)$$

for AdaBoost learned boundary. In addition,

$$ED(\mathbf{x}, q) = \|\mathbf{x} - q\|_2 \quad (6)$$

is the Euclidean distance between image $\mathbf{x}$ and the query q. While M is the maximum Euclidean distance among the positive images to the query,

$$M = \max_{\mathbf{x}} ED(\mathbf{x}, q),\ \forall\, D(\mathbf{x}, \Theta) \geq 0. \quad (7)$$

$M - D(\mathbf{x}, \Theta)$ can be viewed as a kind of *pseudo Euclidean distance* measure for ranking any negative image $\mathbf{x}$.

ISBN 0-7695-1272-0/01 $10.00 (C) 2001 IEEE

5. Experiments

Our restricted similarity measure is evaluated on a subset of Corel photo Gallery. We select 10,009 images with ground truth of 79 concepts or classes. Recall and precision are used to evaluate the retrieval performance. Recall is the ratio of the number of relevant images returned to the total number of relevant images. Precision is the ratio of the number of relevant images returned to the total number of images returned. We calculate precision and recall with respect to the number of relevance feedback. The results of the traditional Euclidean distance measure are given as a baseline in the evaluation. Note that although retrieval results based on Euclidean distance measure are shown in the same figure, there is no feedback (no learning, or we call no restriction) to it. The curves for the Euclidean distance measure are drawn with respect to the number of displayed images, which equals $40 \times R$ in our experiments, with R the number of relevance feedback.

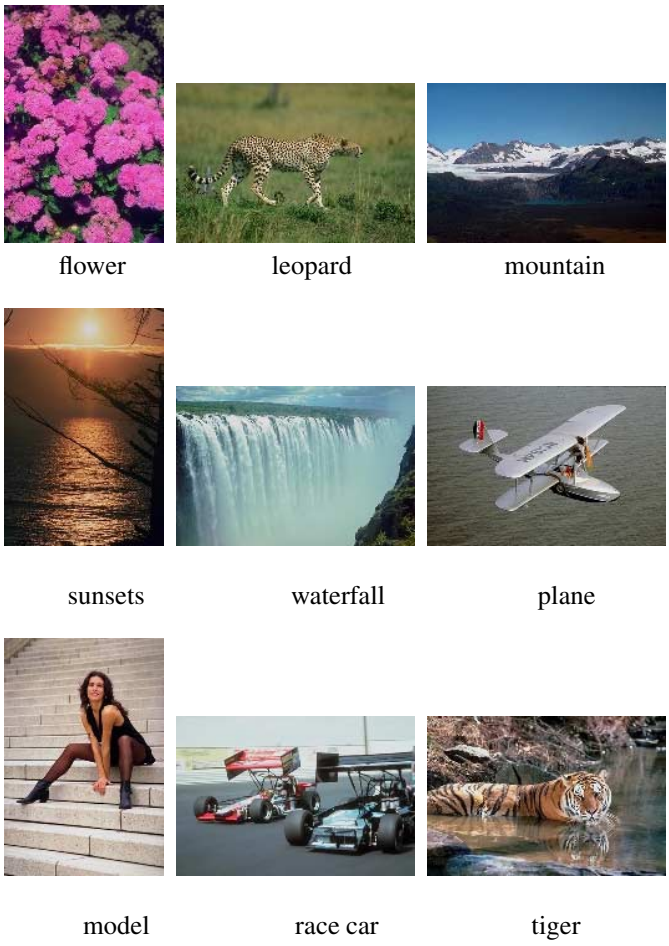


Figure 2. Representative images of 9 groups.

5.1. Image Data Set

The Corel Gallery contains 1,000,000 images, and uses semantic concepts to group them, each group with 100 images. However, their divisions can not be used directly as the ground truth to evaluate CBIR algorithms. First, some images have the same or similar content but divided into different directories, such as "Ballon1" and "Ballon2", "Cuisine" and "Cuisines", and so on. We put them into the same group. Second, some "concepts" are abstract and the corresponding images within those groups vary largely in content, for example, "Spring", "Winter", "Hongkong", and "Montreal". It is difficult for current CBIR algorithms to deal with. Therefore, we exclude those groups in our image database. Considering these aspects, we construct a database of 10,009 images of 79 groups. The number of images within each group ranges from 100 to 300.

5.2. Retrieval Performance

Two goals in our experiments: 1) evaluate whether the restricted similarity measure can deliver better retrieval results; 2) compare to see which method (SVM or AdaBoost) leads to a better performance for filtering.

We select 9 concepts out of 79 to evaluate the retrieval performance, *i.e.*, "flower" (200), "leopard" (100), "model" (300), "mountain" (200), "plane" (200), "race car" (209), "sunsets" (200), "tiger" (100), and "waterfall" (100), as shown in Fig. 2. The numbers indicate how many images in each group.

We simulate the user's behavior in relevance feedback as follows: 40 images are shown each time, and the user clicks all the similar images to submit positive response. However, the users typically do not like to click on so many negative examples frequently, they may just click on the negative in the first round. In the precision and recall curves, the total feedback times are 9, with 0 feedback referring to the retrieval based on Euclidean distance measure without interaction. The boundary is updated continuously afterwards.

For each concept, the precision and recall are averaged over all query images, which is a more representative evaluation. Filtering based on the boundaries learned by SVMs can usually deliver a better result in comparison with that learned by AdaBoost, such as in retrieval of "flower", "leopard", "mountain", "waterfall", "plane", "race car", and "tiger". The AdaBoost based boundary learning can only present performance close to the SVM based approach for "sunsets" and "model", but still worse than that based on SVM. Furthermore, the worst cases for AdaBoost approach are in the retrieval of "race car" and "tiger", in which the boundary restrictions do not improve or only improve marginally over the Euclidean distance measure. Due to space limitations, these separate results are

ISBN 0-7695-1272-0/01 $10.00 (C) 2001 IEEE

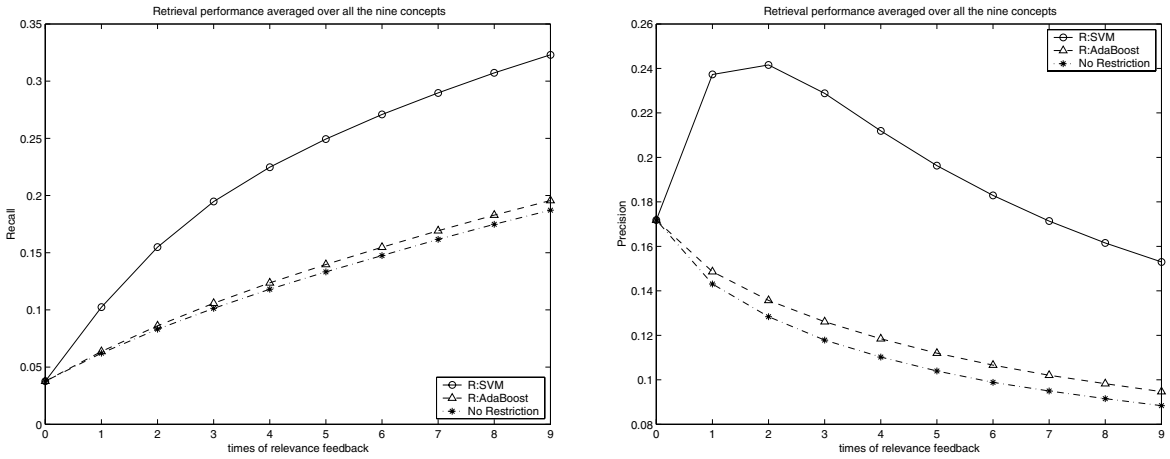


Figure 3. Averaged precision and recall versus the number of relevance feedback of R:SVM, R:AdaBoost, and No Restriction, over the total nine concepts.

not shown here. In stead, only the whole precision and recall curves averaged over the selected 9 concepts (of 1609 queries) is given in Fig. 3. It is obvious that both precision and recall are explicitly improved by using the boundary (learned with SVM) restricted similarity measure. Even only after one or two feedback, the performance has dramatically improved. However, the averaged performance of AdaBoost has marginal improvement over the Euclidean distance measure.

5.3. Discussions

In our RSM with SVM, we use all the 435 features. A further consideration is to reduce the feature dimensionality so as to speed up the retrieval process. In AdaBoost [23], feature selection is incorporated into the learning stage. Usually 20 rounds of boosting is enough, and hence 20 features are used. We would like to see if a similar method can be used for SVM to simply select a small number of features. For this purpose, we try a simple method for feature selection for SVM, similar to that in [23] for AdaBoost. In Fig. 4, we show the averaged precision and recall performance over 200 images of "flower", with $m = 20$ features selected and used for SVM, denoted as "R:SVM-20" for simplicity. It is obvious that its performance is worse than the traditional Euclidean distance metric. To see whether it is because the number of features is too small, we let $m = 50$ and $m = 100$ and show the results in the same figure. The performances of "R:SVM-50" and "R:SVM-100" are worse than the AdaBoost based approach, which indicates the major problem is not the number of selected features. The simple feature selection method can not be used for SVM, while a more elaborated algorithm [26] can be tried instead. This also indirectly indicates the different mechanism for SVM and AdaBoost.

The number of support vectors is determined automatically in SVM learning. If too many examples are presented in feedback (although not actual in practice), the SVM may use lots of support vectors, which makes the filtering process slow. Some methods [19] can be tried to reduce the number of support vectors.

Finally, the AdaBoost filtering may have better results than the SVM for some individual queries. In such cases, how to select the better or combine them to deliver a good result is an open question.

6. Conclusions

We have presented a restricted similarity measure (RSM) for content based image retrieval. This measure takes into consideration the perceptual similarity between images and improves the retrieval performance. Two techniques are used to learn the boundary, and the experimental results indicate that generally the SVM based method is much better than the AdaBoost based approach.

References

[1] C. Cortes and V. Vapnik, Support vector networks, *Machine Learning*, 20, 273-297, 1995.

[2] R. P. W. Duin, Classifiers in almost empty spaces, *Proc. of Internal Conf. on Pattern Recognition*, vol. 2, 1-7, 2000.

[3] Y. Freund and R. E. Schapire, A decision-theoretic generalization of online learning and an application to boosting. *J. Comp. & Sys. Sci.*, 55(1):119-139, 1997.

[4] J. Huang, S. R. Kumar, and M. Metra, Combining supervised learning with color coorelograms for content-based image retrieval. *Proc. of ACM Multimedia'95*, 325-334, 1997.

ISBN 0-7695-1272-0/01 $10.00 (C) 2001 IEEE

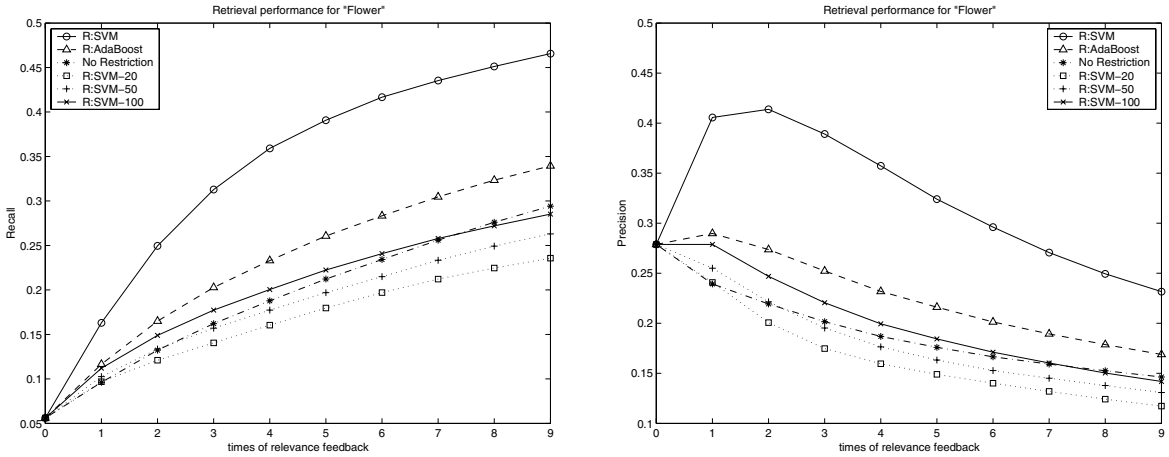


Figure 4. Averaged over 200 queries of "flower" to evaluate the simple feature selection for SVM (R:SVM-m), compared with R:SVM, R:AdaBoost, and the Euclidean, for $m = 20, 50$, and 100.

[5] B. Johansson, A survey on: contents based search im image databases, http://www.isy.liu.se/cvl/Projects/VISIT-bjojo/, Feb. 15, 2000.

[6] W. Y. Ma and B. S. Manjunath, Texture features and learning similarity. *Proc. CVPR*, 425-430, 1996.

[7] W. Y. Ma and H. J. Zhang, Content-based image indexing and retrieval, Chapter 11, *In Handbook of Multimedia Computing*, Borko Furht, ed. CRC Press, 1998.

[8] B. S. Manjunath and W. Y. Ma, Texture features for browsing and retrieval of image data. *IEEE PAMI*, 837-842, 1996.

[9] S. G. Mallat, A theory for mutiersolution signal decomposition: the wavelet representation, *IEEE Trans. on Patern Analysis and Machine Intelligence*, vol 11, 674-693, July 1989.

[10] T. P. Minka and R. W. Picard, Interactive learning using a "society of models", Technical Report No. 349, MIT Media Laboratory, 1995.

[11] W. Niblack, *et al*, The QBIC project: querying images by content using color, texture, and shape, *Proc. of SPIE*, SRIVD, v. 1908, San Jose, CA, 173-187, Feb. 1993.

[12] G. Pass and R. Zabih, Histogram refinement for content-based image retrieval, IEEE Workshop on Applications of Computer Vision, 96-102, 1996.

[13] A. Pentland, R. W. Picard, and S. Sclaroff, Photobook: tools for content based manipulation of image databases, *Proc. of SPIE*, SRIVD, No. 2185, San Jose, CA, 34-47, Feb. 1994.

[14] J. Puzicha, J. M. Buhmann, Y. Rubner and C. Tomasi, Emperical evaluation of dissimilarity measures for color and texture, *Proc. ICCV*, vol. *II*, 1165-1172, 1999.

[15] Y. Rui, et al, A relevance feedback architecture in content-based multimedia information retrieval systems, *Proc. of IEEE Workshop on CAIVL*, 1997.

[16] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, Relevance feedback: a powerful tool for interactive content-based image retrieval, *IEEE Trans. on CSVT*, No. 5, 644-655, 1998.

[17] S. Santini and R. Jain, Similarity measures, *IEEE PAMI*, vol. 21, No. 9, 871-883, 1999.

[18] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, Content-based image retrieval at the end of the early years, *IEEE PAMI*, vol. 22, No. 12, 1349-1380, Dec. 2000.

[19] B. Scholkopf, *et al.*, Input space versus feature space in kernel-based methods, *IEEE Trans. on Neural Networks*, Vol. 10, No. 5, 1000-1017, Sept. 1999.

[20] M. Stricker and M. Orengo, Similarity of color images, SPIE Storage and Retrieval for Image and Video Databases III, vol 2185, 381-392, Feb. 1995.

[21] M. J. Swain and B. H. Ballard, Color indexing, *Int'l J. Computer Vision*, vol, 7, No. 1, 11-32, 1991.

[22] H. Tamura, S. Mori, and T. Yamawaki, Texture features corresponding to visual perception, *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 8, no. 6, 1978.

[23] K. Tieu and P. Viola, Boosting image retrieval, in *Proc. of CVPR*, v. 1, 228-235, 2000.

[24] A. Vailaya, M. A. T. Figueiredo, A.K. Jain and H.J. Zhang, Image Classification for content-based indexing, *IEEE Trans. Image Processing*, v. 10, no. 1, 117-130, 2001.

[25] V. N. Vapnik, *Statistical learning theory*, John Wiley & Sons, New York, 1998.

[26] J. Weston, *et al*, Feature selection for SVMs, NIPS, vol. 13, 2000.

[27] G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae.*, John Wiley and Sons, New York, NY, 1982.

ISBN 0-7695-1272-0/01 $10.00 (C) 2001 IEEE