ADABOOST IN REGION-BASED IMAGE RETRIEVAL*

Sheng-Yang Dai and Yu-Jin Zhang

Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
dai_sy@mails.tsinghua.edu.cn, zhangyj@ee.tsinghua.edu.cn

ABSTRACT

In this paper, a region-based AdaBoost (RBA) algorithm that combines the similarity contributions from different regions in images to form a single value for measuring similarity between images is proposed. The region-based framework utilizes the segmentation result to capture the higher-level concept of images. AdaBoost is a method of finding a highly accurate classifier by combining weak classifiers. A modified version of AdaBoost which can get confidence-rated prediction is applied to learn the final similarity function from user's feedback. It is based on a novel selection of weak classifiers. Experimental and comparison results, which are performed using a general-purpose database containing 7,000 images, are promising.

1. INTRODUCTION

To improve the performance of content-based image retrieval (CBIR) systems, the spatial information in the image should be preserved and employed. Segmentation can help representing images at object-level to express the spatial information properly. Region-based retrieval systems utilize these segmentation results.

Recent research has put attention on how to integrate the similarity contributed by various regions together to form a single value. For example, in Blobworld [1], user needs to specify blobs to tell the system the interested objects. Wang *et al.* [2] proposed integrated region matching (IRM) to automatically get the similarity that combines information from all regions. Each region is assigned a weight that is proportional to its size. However, this does not necessarily reflect the user's intention and the large background regions may have undesirable large weights. Jing *et al.* [3] proposed a retrieval method called self-learned region importance (SLRI), which is better than IRM in retrieval performance. It is based on the assumption that the similarity between images is the linear summation of the contributions from all regions.

AdaBoost [4] is a recently developed learning algorithm. It boosts the classification performance by combining a collection of weak classifiers to form a stronger classifier. In each step of AdaBoost, the classifier with the best performance is selected and a higher weight is put on the miss-classified training data. In this way, the classifier will gradually focus on the difficult examples to be classified correctly. In theory, it is proved that AdaBoost could minimize the margin between positive and negative examples. An AdaBoost algorithm which can get confidence-rated prediction is proposed in [5]. It has been successfully applied in [6].

AdaBoost has been introduced into CBIR systems recently. One of the most impressive approaches was presented by Tieu and Viola [7]. About 45,000 features are produced, most of which only reacted well with a small subsets of images. AdaBoost is applied to select the features that can best separate a small positive sample set of images from a slightly larger set of negative examples. However, this system is only based on low-level features without performing image segmentation. Therefore, the selected features do not have the capability to reflect higher-level concepts, which could be captured by region-based algorithm.

The region-based AdaBoost (RBA) framework is developed in this paper to integrate AdaBoost algorithm together with region-based representation of images.

The remainder of the paper is organized as follows. The proposed algorithm is described in section 2, the experimental results are presented in section 3, and some concluding remarks are given in section 4.

2. REGION-BASED ADABOOST

2.1. Similarity and Mapping

The segmentation method we use is described in [8]. For each image I, two sets of parameters are used to get two level descriptions: detailed description $\{R_1^d, R_2^d, \ldots, R_m^d\}$ and rough description $\{R_1^r, R_2^r, \ldots, R_n^r\}$. The query image I_Q is described by a rough description $\{R_{Q,1}^r, R_{Q,2}^r, \ldots, R_{Q,m}^r\}$, while each candidate image I_C (assume it is composed by

* This work has been supported by NSFC Project 60172025.

n regions) is described by a detailed description $\{R^d_{C,1}, R^d_{C,2}, \ldots, R^d_{C,n}\}$. The two-level description scheme is designed to limit the influence of inaccurate segmentation and make the region mapping between query and candidate images to become an one-to-many mapping.

Similarity between regions R_1 and R_2 is defined as $s(R_1, R_2) = s_{color}(R_1, R_2) \times s_{shape}(R_1, R_2)$, where $s_{color}(R_1, R_2)$ is the similarity between the average colors of the two regions, and $s_{shape}(R_1, R_2)$ is the overlapped size of the two regions based on the grid partition of entire images.

A mapping function is defined to map each region in the candidate image I_C to one region in the query image.

$$Map(k) = \arg\max_t s(R^r_{Q,t}, R^d_{C,k}) \qquad (1)$$

The similarity contribution of each region in the query image is defined as follows:

$$f_{C,i} = \frac{s(R^r_{Q,i}, I_C)}{size(R^r_{Q,i})} = \frac{\sum_{1 \le k \le n, Map(k)=i} s(R^r_{Q,i}, R^d_{C,k})}{size(R^r_{Q,i})} \qquad (2)$$

where $s(R^r_{Q,i}, I_C)$ is the similarity between $R^r_{Q,i}$ and the candidate images, it is the summation of similarities between $R^r_{Q,i}$ and $R^d_{C,k}$ that is mapped to $R^r_{Q,i}$. $size(R)$ is the size of region R. It works as a normalization factor so that $f_{C,i} \in [0, 1]$.

Each candidate image is described by an m-dimensional feature vector:

$$\boldsymbol{F}_C = (f_{C,1}, f_{C,2}, \ldots, f_{C,m}) \qquad (3)$$

Assume $I_1^+, I_2^+, \ldots, I_s^+$ and $I_1^-, I_2^-, \ldots, I_t^-$ are the positive and negative examples in a feedback procedure. $\boldsymbol{F}_i^+ = (f^+_{i,1}, f^+_{i,2}, \ldots, f^+_{i,m})$, $\boldsymbol{F}_j^- = (f^-_{j,1}, f^-_{j,2}, \ldots, f^-_{j,m})$, $i = 1, 2, \ldots, s$, $j = 1, 2, \ldots, t$ are the correspondent feature vectors for the example images. AdaBoost is followed to get the function $P(\boldsymbol{F})$ describing the likelihood that a feature vector $\boldsymbol{F}$ is positive through learning from the above feature vectors.

In the practical retrieval procedure of the proposed framework, a summation of $f_{C,i}$ weighted by the region size is computed firstly for the similarities, and this is the result returned to the user in the first round. Assume that there are s correct results in the first T_{num} retrieved images, and they are set to be the positive examples. In our experiments, $T_{\text{num}} = 100$ is used in considering both of the user tolerance and the plenitude of training data. The number of negative examples is $t = \alpha s$, in which $min(T_{\text{num}} - s, \beta s)$, $(\beta < \alpha)$ of them are randomly selected from the negative images in the first T_{num} images, and the others are randomly selected from the entire image database. Experiments are performed based on various choices of α and β, and it is shown that their values do not affect the performance much. In our experiments, $\alpha = 4$, $\beta = 2$ are used, which means that the number of negative examples is 4 times of that of positive examples, and in the general case, half of them are chosen from the first T_{num} result. The reason for selecting negative examples as above is that it can enable AdaBoost to learn the ability to discriminate positive examples from both images with large region similarity and general images.

2.2. Learning Weak Classifier

Each of the weak classifier is based on one component of the feature vector. Assume $f_1^+, f_2^+, \ldots, f_s^+, f_1^-, f_2^-, \ldots, f_t^-$ are the values of the current component of all the training data, with correspondent weight $w_1^+, w_2^+, \ldots, w_s^+, w_1^-, w_2^-, \ldots, w_t^-$. $p^+(\boldsymbol{F}_C) = p^+(f)$ is defined as the likelihood that $\boldsymbol{F}_C$ is the feature vector for a positive example (f is the value of the current component of $\boldsymbol{F}_C$).

It is naturally to assume that a larger f leads to a larger likelihood, so the following continuous piecewise linear function is adopted:

$$p^+(f) = p^+_{T_m,T_p,T_n}(f) = \begin{cases} 1 & T_p < f \le 1 \\ 0.5 + 0.5 \times \dfrac{f - T_m}{T_p - T_m} & T_m < f \le T_p \\ 0.5 & f = T_m \\ 0.5 - 0.5 \times \dfrac{f - T_m}{T_n - T_m} & T_n \le f < T_m \\ 0 & 0 \le f < T_n \end{cases} \qquad (4)$$

$p^-(\boldsymbol{F}_C) = p^-(f)$ is defined as the likelihood that $\boldsymbol{F}_C$ is a negative example (f is value of the current component).

$$p^-(f) = 1 - p^+(f) \qquad (5)$$

The parameters T_m, T_p, and T_n are determined by the following functions:

$$T_m = \arg\min_{thr}\Big(\sum_{1 \le i \le s, f_i^+ < thr} w_i^+ + \sum_{1 \le j \le t, f_j^- > thr} w_j^-\Big) \qquad (6)$$

$$T_p = \arg\min_{thr}\Big\{\sum_{\substack{1 \le i \le s, \\ f_i^+ > T_m}} [w_i^+ \times p^-_{T_m,thr,0}(f_i^+)] + \sum_{\substack{1 \le j \le t, \\ f_j^- > T_m}} [w_j^- \times p^+_{T_m,thr,0}(f_j^-)]\Big\} \qquad (7)$$

$$T_n = \arg\min_{thr}\Big\{\sum_{\substack{1 \le i \le s, \\ f_i^+ < T_m}} [w_i^+ \times p^-_{T_m,1,thr}(f_i^+)] + \sum_{\substack{1 \le j \le t, \\ f_j^- < T_m}} [w_j^- \times p^+_{T_m,1,thr}(f_j^-)]\Big\} \qquad (8)$$

The underlying physical concept of above functions is intuitive. T_m is defined to minimize the classification error of training data, and T_p, T_n are defined to make the functions best fit the training data (in the meaning of minimizing the error probability).

2.3. AdaBoost Procedure

The AdaBoost algorithm that can get confidence-rated prediction [5] is utilized in our framework for image retrieval. The entire procedure is described in the following four steps:

(1) Given feature vectors for $s + t$ training data: $\boldsymbol{F}_1^+$, $\boldsymbol{F}_2^+$, …, $\boldsymbol{F}_s^+$ are the positive feature vectors and $\boldsymbol{F}_1^-$, $\boldsymbol{F}_2^-$, …, $\boldsymbol{F}_t^-$ are the negative feature vectors.

(2) Weight initialization: the summation for positive examples and the summation for negative examples are both equal to 0.5. All positive examples and all negative examples are assigned equal weights respectively. So we have $w_{1,i}^+ = 1/2s, w_{1,j}^- = 1/2t$, $i = 1, 2, \ldots, s$, $j = 1, 2, \ldots, t$.

(3) For $k = 1, 2, \ldots, K$, get the likelihood function $p_k^+(f)$ with the following steps:

i) Train one weak classifier for each component of the feature vector with current weight.

ii) Choose the weak classifier with the lowest classification error. Assume its correspondent likelihood function is $p_k^+(f)$, and l_k is the index for current component, then, the classification error is:

$$\varepsilon_k = \sum_{1 \le i \le s, p_k^+(F_i^+) < 0.5} w_i^+ + \sum_{1 \le j \le t, p_k^+(F_j^-) > 0.5} w_j^- \tag{9}$$

The weight for the k-th weak classifier is:

$$\alpha_k = \ln \frac{1-\varepsilon_k}{\varepsilon_k} \tag{10}$$

iii) Update the weights for training data by the following functions:

$$w_{k+1,i}^+ = \frac{1}{Z^+} w_{k,i}^+ \exp(-\frac{1}{2}\alpha_k \ln(\frac{p_k^+(f_{i,l_k}^+)+\delta}{p_k^-(f_{i,l_k}^+)+\delta}) \tag{11}$$

$$w_{k+1,j}^- = \frac{1}{Z^-} w_{k,j}^- \exp(\frac{1}{2}\alpha_k \ln(\frac{p_k^+(f_{j,l_k}^-)+\delta}{p_k^-(f_{j,l_k}^-)+\delta}) \tag{12}$$

where Z^+ and Z^- are the normalization factors to make the weight summations for both positive and negative training data remain 0.5. It can be seen that larger weights are assigned to examples that are wrongly classified and smaller weight to examples that are correctly classified.

(4) The final likelihood that feature vector $\boldsymbol{F}$ belongs to a positive example is:

$$P(\boldsymbol{F}) = \tanh[\sum_{k=1}^{K} \alpha_k \ln \frac{p_k^+(\boldsymbol{F})+\delta}{p_k^-(\boldsymbol{F})+\delta}] \tag{13}$$

tanh function is used to map the value into [-1, 1]. δ is added in Eqs. (11), (12), (13) to avoid numerical problems when $p_k^+(\boldsymbol{F})$ or $p_k^-(\boldsymbol{F})$ is very small or even zero. In addition, overly confident predictions tend to cause overfit [5]. $\delta = 0.1$ is taken in our experiment. The final result after feedback is ordered by the value of $P(\boldsymbol{F})$.

In the above procedure, the number of weak classifier K is related to the region number of images. $K = 7$ is used in our experiments, which has shown the best performance.

3. EXPERIMENTAL COMPARISONS

The image database used for performance testing contains 7,000 general-purpose images from Corel photo collection. These images have been already classified into 70 different classes with 100 images in each class. Generally speaking, the images in each class have some comparable semantic meanings and also quite similar visual perception. We take this original classification as the ground truth for judging our statistical retrieval result. In the experiments, the user's feedback is also simulated with the help of this classification. Images in the same class as the query image in the first T_{num} retrieval result are set to be the positive examples selected by user.

The comparison results are shown in Fig.1, in which the average precision curves are plotted. The average precision value, noted by $P'(n)$, is obtained by taking the first n retrieval results. These results are based on the retrieval experiments with 700 (10 in each class) query images selected randomly from the 7000 images mentioned above. In Fig.1, three methods are compared. In the 1st method, the similarity value is taken as the weighted summation (WS) of the similarity contribution of all regions in the query image, and the weights are equal to the region size. In the 2nd method, the similarity value is taken as the summation of the similarity contribution of all regions with weights computed by the technique in [3] (SLRI). The 3rd one is obtained by using the proposed region-based AdaBoost (RBA) framework. It can be seen that SLRI is a little better than WS, and the proposed algorithm RBA outperforms the other two methods. As SLRI has a higher retrieval performance than IRM [3], so the proposed algorithm RBA also outperforms the performance of IRM.

The top 20 images found by WS and the proposed RBA algorithm are shown in Fig. 2 and Fig. 3, respectively. The query image is the left-up image in each group. It can be seen that in Fig. 2, background regions (grass and road in (a), green wall in (b)) play over-important roles, thus the result is poor. After the region-based AdaBoost learning procedure, semantically important regions (such as car in (a), pumpkin in (b)) are learned by AdaBoost and play significant roles in the retrieval process. The final results are improved.

4. CONCLUDING REMARKS

A novel image retrieval framework based on region-based AdaBoost is proposed in this paper. It can combine similarity contribution computed from each region of the query image. Experiments based on an images database with 7,000 general-purpose and randomly selected query images shows the effectiveness of the proposed techniques.

5. REFERENCES

[1] C. Carson et al. "Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying", *IEEE Trans. PAMI*, vol. 24(8), pp. 1026-1038, 2002.

[2] J. Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries", *IEEE Trans. PAMI*, vol. 23(9), pp. 947-963, 2001.

[3] F. Jing, B. Zhang, F Lin, W. Y. Ma, and H. J. Zhang, "A Novel Region-Based Image Retrieval Method Using Relevance Feedback", *Proc. 3rd ACM Intl Workshop on Multimedia Information Retrieval (MIR)*, 2001.

[4] R. E. Schapire, "A Brief Introduction to Boosting", *Proc. of Sixteen Int. Joint Conf. on Artificial Intelligence*, pp. 1401-1405, 1999.

[5] R.E. Schapire and Y. Singer. "Improved Boosting Algorithms Using Confidence-Rated Predictions", *Machine Learning*, vol. 37(3), pp. 297-336, 1999.

[6] C. Liu and H. Y. Shum, "Kullback-Leibler Boosting", *Proc. CVPR*, pp. 587-594, 2003.

[7] K. Tieu and P. Viola "Boosting Image Retrieval", *Proc. CVPR*, pp. 228-235, 2000.

[8] S. Y. Dai and Y. J. Zhang, "Color Image Segmentation with Watershed on Color Histogram and Markov Random Fields", *Proc. of ICICS-PCM*, p0240, 2003.

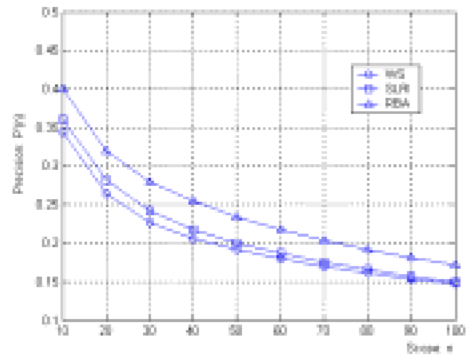
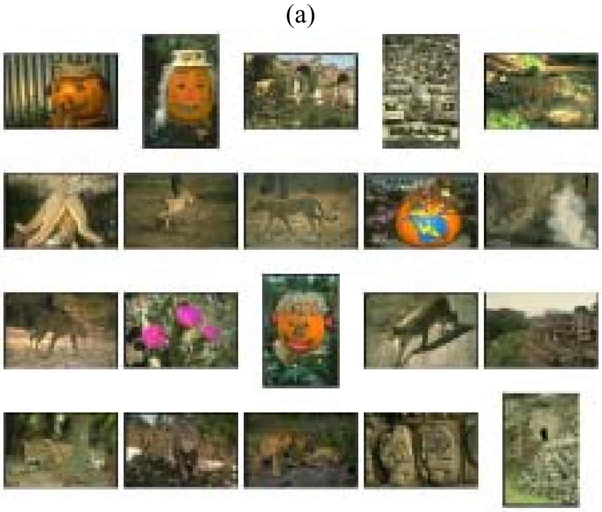


Fig. 1: Average precision curve

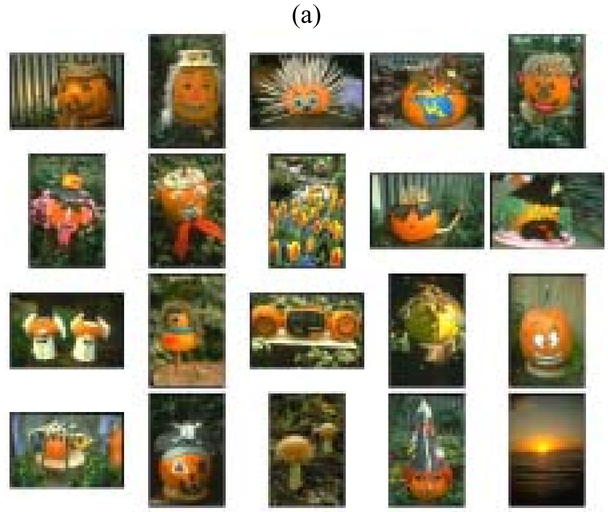
(a)



(b)

Fig. 2: Retrieval results obtained by using WS

(a)



(b)

Fig. 3: Retrieval results obtained by using RBA