Multimed Tools Appl (2007) 35:71–89
DOI 10.1007/s11042-007-0116-9

OCRS: an interactive object-based image clustering and retrieval system

Chengcui Zhang · Xin Chen

Published online: 2 May 2007

Abstract In this paper, we propose an Interactive Object-based Image Clustering and Retrieval System (OCRS). The system incorporates two major modules: Preprocessing and Object-based Image Retrieval. In preprocessing, an unsupervised segmentation method called WavSeg is used to segment images into meaningful semantic regions (image objects). This is an area where a huge number of image regions are involved. Therefore, we propose a Genetic Algorithm based algorithm to cluster these image objects and thus reduce the search space for object-based image retrieval. In the learning and retrieval module, the Diverse Density algorithm is adopted to analyze the user's interest and generate the initial hypothesis which provides a prototype for future learning and retrieval. Relevance Feedback technique is incorporated to provide progressive guidance to the learning process. In interacting with user, we propose to use One-Class Support Vector Machine (SVM) to learn the user's interest and refine the returned result. Performance is evaluated on a large image database and the effectiveness of our retrieval algorithm is demonstrated through comparative studies.

Keywords OCRS · Object-based image retrieval · SVM · Genetic Algorithm · Clustering

1 Introduction

With the rapid increase of digital image data, image retrieval has drawn attention of researchers in computer vision and database communities. However, the current

C. Zhang (✉) · X. Chen
Department of Computer and Information Sciences, University of Alabama at Birmingham, Birmingham, AL 35294-1170, USA
e-mail: Zhang@cis.uab.edu

X. Chen
e-mail: Chenxin@cis.uab.edu

state-of-art technologies are facing two main problems: (1) "semantic gap" between low level features and high level concept; (2) curse of dimensionality. This paper aims to build a framework to alleviate the above problems especially for object-based image retrieval. In this paper, an object refers to an image region.

For "semantic gap" problem, Relevance feedback (RF) technique is widely used to incorporate the user's concept with the learning process [21, 23]. As a supervised learning technique, it has been shown to significantly increase the retrieval accuracy. However, most of the existing RF-based approaches consider each image as a whole, which is represented by a vector of N dimensional image features. However, the user's query interest is often just one part of the query image i.e. a region in the image that has an obvious semantic meaning. Therefore, rather than viewing each image as a whole, it makes more sense to view it as a set of semantic regions. In this context, the goal of image retrieval is to find the semantic region(s) of the user's interest.

In region-based CBIR, each image is composed of several semantically meaningful regions. If we consider each image as a bag and each region as an instance, an image is then a bag of instances. In this way, region-based CBIR is transformed into a Multiple Instance Learning (MIL) problem. Maron et al. applied MIL into natural scene image classification [18]. In the scenario of MIL, the labels of individual instances in the training data are not available, instead the bags are labeled. When applied to RF-based image retrieval, this corresponds to the scenario that the user gives feedback on the whole image (bag) although he/she may be only interested in a specific region (instance) of that image. The goal of MIL is to obtain a hypothesis from the training examples that generates labels for unseen bags (images) based on the user's interest in a specific region. In other words, the instance labels have to be derived from the bag labels, based on the following two properties of MIL: (1) if the bag label is positive, there exists at least one instance in the bag that is positive; (2) if the bag label is negative, all the instances in the bag are negative.

In order to support region-based image retrieval, we need to divide each image into several semantic regions. Instead of viewing each image as a whole, we examine region similarity during image retrieval. However, this further increases the search space by a factor of 4–6. Clustering is a process of grouping a set of physical or abstract objects into classes based on some similarity criteria. In this study, the objects correspond to image regions. Given the huge amount of semantic regions in this problem, we first preprocess image regions by dividing them into clusters. In this way the search space can be reduced to a few clusters that are relevant to the query region. K-means is a traditional clustering method and has been widely used in image clustering such as [15, 25]. However, it is incapable of finding non-convex clusters and tends to fall into local optimum especially when the number of data objects is large. In contrast, Genetic algorithm [11] is known for its robustness and ability to approximate global optimum. In this study, we adapted this algorithm to suit our needs of clustering image regions.

After clustering, our proposed system applies Diverse Density (DD) as proposed within the framework of MIL by Maron et al. [18] to learn the region of interest from the user's relevance feedback on the whole image and then shifts its focus of attention to that region. In [31], Zhang et al. further extends Maron's work by incorporating EM (Expectation-Maximization) algorithm. We adopt Zhang's method because it is less sensitive to the dimension of feature space and scales up well. We take the output

of DD as our initial hypothesis of user's interest and continue the feedback-retrieval process using our proposed kernel learning algorithm.

In the following learning procedure, we use One-Class Support Vector Machine (SVM) to learn from user's feedback and refine the retrieval result. The motivation comes from the fact that positive samples are all alike, while negative samples are each bad in their own way. In other words, instead of building models for both positive class and negative class, it makes more sense to assume that all positive regions are in one class while the negative regions are outliers of the positive class.

In our approach, One-Class SVM is used to model the non-linear distribution of image regions and to separate positive regions from negative ones. Each region of the test image is given a score by the evaluation function built from the model. The higher the score, the more similar it is to the region of interest. The images with the highest scores are returned to the user as query results. However, the critical issue here is how to transform the traditional SVM learning, in which labeled training instances are readily available, to a MIL learning problem where only the labels of bags (e.g. images with positive/negative feedbacks) are available. In this study, we proposed a method to solve the aforementioned problem and our experiments show that high retrieval accuracy can be achieved usually within 4 iterations.

Section 2 gives a brief literature review. In Section 3, an overview of our OCRS system is presented. The preprocessing module is detailed in Section 4 which involves segmentation and clustering. The detailed learning and retrieval approach is discussed in Section 5. In Section 6, the evaluation of system performance with experimental results is presented. Section 7 concludes the paper.

2 Related work

Numerous works have been done in the area of Content-Based Image Retrieval (CBIR). Basically, they can be divided into two categories: region-based and non-region based. A well-known system of region-based CBIR is Blobworld [2]. In this system, a simple distance-based method is used to measure the similarity between image regions. There is no learning mechanism involved. In our approach, by learning from the user's relevance feedback [21, 23], we can refine the retrieval result using a more sophisticated learning algorithm. Therefore, our algorithm falls into the category of region based CBIR with learning capabilities. Some other works in this category can be found in [5, 12, 13, 14].

2.1 Image clustering

Clustering is a well-studied topic in the area of data mining. In its application to image data, algorithms such as K-means [15, 25], K Nearest Neighbor [9, 27] have gained great popularity. Krishnamachari et al. use a hierarchical clustering algorithm for image retrieval [17]. Kim et al. designed a clustering algorithm by incorporating relevance feedback to generate dynamic clusters [16]. Chen et al. [6] also implemented a cluster-based retrieval system. Their clustering algorithm is based on graph theory.

In our work, the main purpose of clustering is to reduce the effect of the curse of dimensionality in object-based image retrieval, which is a problem that barely any work addresses. With clustered image databases, the search space for object-based image retrieval is greatly reduced. Our proposed mechanism of image region clustering is based on Genetic Algorithm for the reason mentioned in the Introduction Section.

2.2 Multiple instance learning

A great amount of research has been done to solve Multiple Instance Learning problems. A representative approach by learning the axis-parallel rectangles is first developed by Dietterich et al. [8]. The concept of Diverse Density (DD) is introduced by Maron and Lozano-Perez [18] and a two-step gradient descent with multiple starting points is applied to find the maximum Diverse Density. The EM-DD algorithm is proposed by Zhang and Goldman [31] based on Diverse Density. Its main difference from Maron's method is that it searches maximum DD points by Expectation Maximization. It is shown that EM-DD is more robust in dealing with high-dimension data. Wang et al. [26] explore the lazy learning approaches in Multiple Instance Learning. Zucker et al. [32] attempt to solve the Multiple Instance Learning problem with decision trees and decision rules. Ramon et al. [19] propose the Multiple Instance Neural Network. Andrews et al. use Support Vector Machines (SVMs) to solve MIL problem. Their method is called MI-SVM [1].

Some of the above mentioned algorithms have been applied to image classification or image retrieval. The DD method of Maron et al. [18] is applied into natural scene image classification. This is based on the relation between the bag label and the instance label. The learning goal is to find the positive instances in the positive bags. Huang et al. [12] adopt Neural Network based learning algorithm. The EM-DD method of Zhang et al. [31] is used by Chen et al. [5] for image categorization. With the returned result from EM-DD, Chen et al. apply standard SVMs to solve a two-class classification problem. Their approach is called DD-SVM which is a region based image retrieval algorithm. It is claimed in Chen et al.'s work that their approach is different from [1]'s MI-SVM in that DD-SVM defines several features for each bag according to instance prototypes (generated by DD) while MI-SVM only selects one instance to represent the whole positive bag. In [28], a mechanism for image annotation is proposed, which applies the algorithm of Diverse Density to image region selection. Baysian network is then used for classification with the ultimate goal of annotating images.

In our approach, we apply EM-DD to find a prototype of query instance. In another word, we try to "guess" the user's interested region in the whole query image. In this sense, we adopt a similar method as in DD-SVM. While DD-SVM chooses binary SVM as the learning algorithm, we consider grouping all negative regions into one class somewhat inappropriate for the reason mentioned in Introduction. Therefore, after the initial analysis of user's interest by DD, our proposed learning algorithm concentrates on those positive images and uses the learned region-of-interest to evaluate all the other images in the image database. For this purpose, we applied One-Class Support Vector Machine (SVM) [22] to solve the MIL problem

in CBIR. Chen et al. [7] and Gondra et al. [10] also use One-Class SVM in image retrieval. However, it is applied to the image as a whole. An example of region-based image retrieval using One-Class SVM is in the work of Jing et al. [13, 14]. However, One-class SVM is only used as a distribution estimator and again a binary SVM classifier is built for image retrieval.

3 System architecture

Figure 1 shows the architecture of the proposed system. In the preprocessing module, images are segmented into semantic regions, with each represented by a 19-feature vector. A Genetic Algorithm based clustering method is then implemented to cluster these image segments into clusters so that similar image segments are grouped together.

In the initial query, the system first gets the user's query. However, at this point, the system has no clue to the user's interested semantic region. Therefore, a simple Euclidean based similarity comparison is performed to retrieve the initial query results to the user. After the initial query, the user gives feedback to the retrieved images and these feedbacks are examined by Diverse Density trying to analyze user's

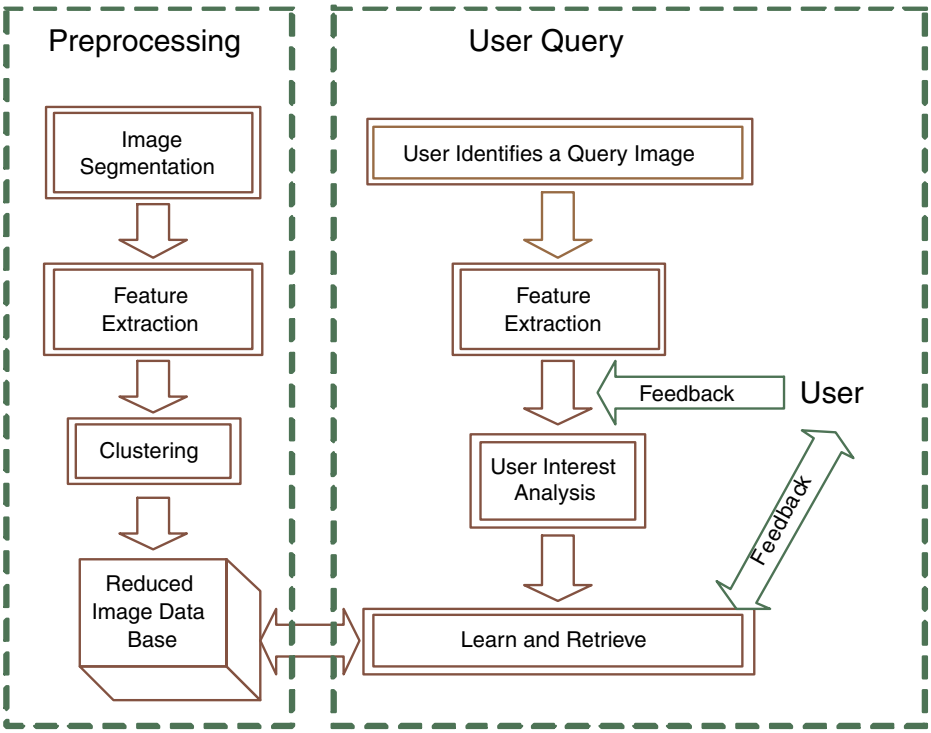


Fig. 1 OCRS system architecture

interest. The output of Diverse Density algorithm is the initial input of One-Class Support Vector Machine (SVM) based algorithm which learns from these feedbacks and starts another round of retrieval. In each round, the refined retrieval result is provided to the user for further feedback. One-Class SVM then studies these feedbacks and builds a model for future retrieval.

4 Preprocessing

4.1 Segmentation

4.1.1 WavSeg image segmentation

Instead of manually dividing each image into a couple of overlapping regions [29], in this study, we propose to use a fast yet effective image segmentation method called WavSeg as proposed in our recent work [3, 30] to partition images. In Wavseg, a wavelet analysis in concert with the SPCPE algorithm [4] is used to segment an image into regions.

By using wavelet transform and choosing proper wavelets (Daubechies wavelets), the high-frequency components will disappear in larger scale sub bands and therefore, the potential regions will become clearly evident. In our experiments, the images are pre-processed by Daubechies wavelet transform because it is proven to be suitable for image analysis. The decomposition level is 1. Then by grouping the salient points from each channel, an initial coarse partition can be obtained and passed as the input to the SPCPE segmentation algorithm. Actually, even the coarse initial partition generated by wavelet transform is much closer to some global minima in SPCPE than a random initial partition. In other words, a better initial partition will lead to better segmentation results. In addition, wavelet transform can produce other useful features such as texture features at the time of extracting the region-of-interest. Therefore, within one entry scanning through the image data, image regions as well as their texture features can be obtained. Based on our initial testing results, the wavelet based SPCPE segmentation framework (WavSeg) outperforms the random initial partition based SPCPE algorithm in average. It is worth pointing out that WavSeg is fast. The processing time for a 240×384 image is only about 0.33 s in average.

Figure 2 provides two groups of examples of the segmentation results. In each row, the first image is the original image. The second one is the segmentation mask file produced by WavSeg that shows all the segments/regions in different colors. The third one is the segmentation mask file of Blobworld [2]. It can be seen from this figure that Blobworld tends to over-segment an image. For example, according to the Blobworld's results, the hawk object in the first example is composed of approximately 4 smaller regions while the deer object in the second example consists of 2 regions, with each corresponding to the upper-body and lower-body of that deer object, respectively. One big problem caused by over-segmenting is the feature extraction for image regions. By way of an example, which part of the deer object—the upper-body or the lower-body—can better represent the deer object? Obviously a combination of these two makes more sense since the user is interested in the entire deer object, not just part of it. However, due to the over-segmenting problem, we

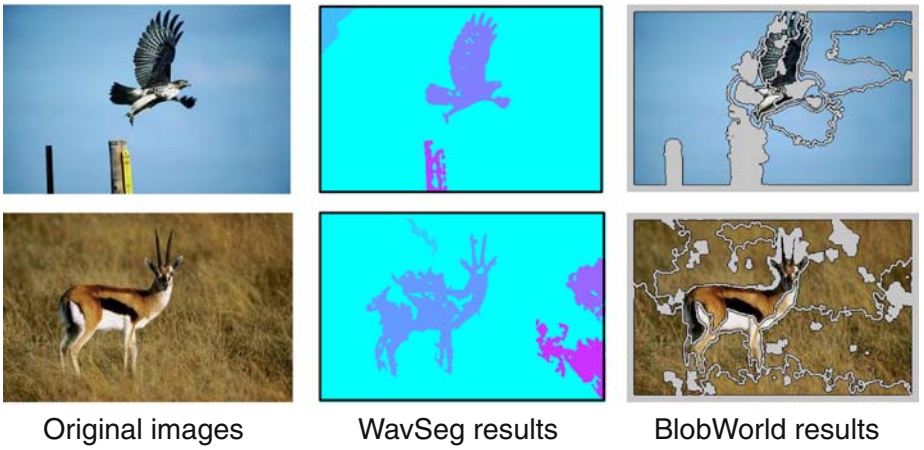


Fig. 2 Sample segmentation result

are not able to obtain the complete information of an object like the deer. On the contrast, it has been shown that, based on our experiments, WavSeg algorithm has advantages over Blobworld in that it is more successful in identifying the "entire" object (segment) instead of partitioning it into several small regions. Therefore, the over-segmenting problem in WavSeg is greatly alleviated.

4.1.2 Region feature extraction

Both the local color and local texture features are extracted for each image region. For color features, HSV color space and its variants are proven to be particularly amenable to color image analysis. Therefore, we quantize the color space using color categorization based on H. S. V. value ranges. Twelve representative colors are identified. They are black, white, red, red-yellow, yellow, yellow-green, green, green-blue, blue, blue-purple, purple, and purple-red. The Hue is divided into five main color slices and five transition color slices. Each transition color slice such as yellow-green is considered in both adjacent main color slices. We disregard the difference between the bright chromatic colors and the chromatic colors. Each transition color slice is treated as a separate category instead of being combined into both adjacent main color slices. A new category "gray" is added so that there are totally thirteen color features for each image region in our method.

For texture features, one-level wavelet transformation using Daubechies wavelets are used to generate four subbands of the original image. They include the horizontal detail sub-image, the vertical detail sub-image, and the diagonal detail sub-image. For the wavelet coefficients in each of the above three subbands, the mean and variance values are collected respectively. Therefore, totally six texture features are generated for each image region in our method.

The 13 color features and six texture features of each region are extracted after image segmentation. Thus, for each image, the number of its objects (regions) is equal to the number of regions within that image. Each object has 19 features.

4.2 Clustering

4.2.1 Overview of genetic algorithm

The basic idea of Genetic Algorithm originates from the theory of evolution – "survival of the fittest." It was formally introduced in the 1970s by John Holland [11]. It is less susceptible to getting "stuck" at local optima. The overview of genetic algorithm is shown in Fig. 3.

The algorithm starts with randomly generating the initial population (possible solutions to a real-world problem). In order to be understood in genetic world, the possible solutions to a real world problem are first encoded. Each solution forms a chromosome. A population is a group of chromosomes. From the first generation, these chromosomes are first evaluated. Then they are operated by three genetic operators: *Selection, Crossover* and *Mutation* and generate the next generation. The next generation of chromosomes is again evaluated. An objective function is used in evaluation which measures the fitness of each individual solution (chromosome). This accomplishes the evolution of the first generation. Genetic algorithm then starts to run the next generation and goes through the above-mentioned process again until an optimal solution is found.

4.2.2 Genetic algorithm design for image region clustering

The objective of image region clustering is to find the optimal combination that minimizes the function below:

$$F(C) = \sum_{i=1}^{n} \min_{j=1}^{k} (d(p_i, C_j)) \tag{1}$$

p_i is an image region in the cluster. C_j is the centroid of cluster j, which is an actual image region but not the virtual center of the cluster. n is the total number of image regions and k is the number of clusters. The value of k is determined experimentally as there is no prior knowledge about how many clusters are there. A too large k value would result in over-clustering and increase the number of false negatives, while a too small k value would not help much in reducing the search space. According to our experiment, in which there are 10,000 images with 49,584 regions, we divide the entire set of image regions into 100 clusters since it results in a good balance between accuracy and efficiency. d is some distance measure. In this study, we use

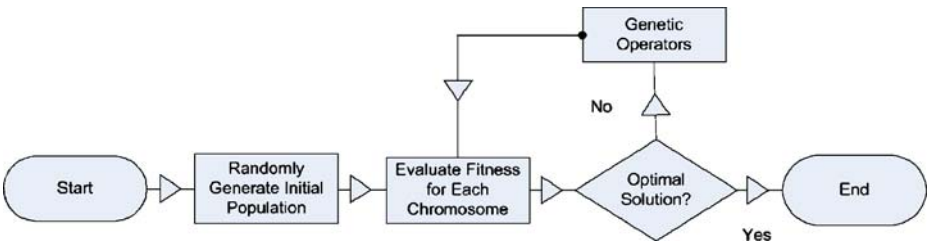


Fig. 3 Genetic algorithm flow chart

the Euclidean distance. Equation 1 is the objective function in our algorithm. The goal is to find its minimum.

In image region clustering, the target is to group semantic image regions into clusters according to their similarities. The above-mentioned representative regions are actually centroids of clusters. Therefore, a feasible solution to a clustering problem would be a set of centroids. To encode it, we give each region an ID: 1, 2, ..., n (n is an integer). The centroids are represented by their IDs in the chromosome.

Figure 4 is an example of a chromosome. In this chromosome, each integer is a gene in genetic world. It corresponds to the ID of a centroid image region in the real world.

The initial size of population is set to l which is 50 in this study. For each chromosome we randomly generate k genes, which are actually k integers between 1 and n (the number of image regions). These k genes correspond to the representative image region for each of the k clusters. We then calculate the inverse values of the objective function for these chromosomes: $f_1, f_2, ..., f_l$. The fitness of each individual chromosome is computed according to (2).

$$Fit_i = f_i / \sum_{i=1}^{l} f_i \tag{2}$$

With the first generation, "evolution" begins. In each generation, the whole population goes through three operators: Selection, Recombination and Mutation.

- *Selection*: There are many kinds of selection operations. We use a Roulette to simulate the selection as shown in Fig. 5. For each chromosome we compute its fitness according to (2). Two chromosomes from the population are then randomly selected. The higher the fitness the higher the chance a chromosome is selected. This mechanism is like rotating roulette as shown in Fig. 5. C_1, C_2, ... are chromosomes. The area each chromosome occupies is determined by its fitness value. Therefore, chromosomes with higher fitness values would have more chances to be selected in each rotation. We select l pairs of chromosomes and feed them into the next step.
- *Recombination*: In this step, the recombination operator proposed in [24] is used instead of a simple crossover. Given a pair of chromosomes C_1 and C_2, we use recombination operator to generate their child chromosome C_0 one gene at a time. A randomly generated small number ρ is used to decide where to get the next gene i.e. from C_1 or C_2 or both. We also want to make sure the chosen gene is not already in C_0. Therefore, each gene in C_0 is either in C_1 or C_2 or both and is not repetitive of other genes in C_0. If we want k clusters, the algorithm has to iterate k times in order to get k genes (centroids) one at a time. The detailed implementation is shown in Fig. 6.
- *Mutation*: In order to obtain high diversity of the genes, a "newly-born" child chromosome may mutate one of its genes to a random integer between 1 and n. However, this mutation is operated at a very low frequency and it varies dynamically through generations.

Fig. 4 An example of chromosome

98	56	10	23	65	35	22	469	16

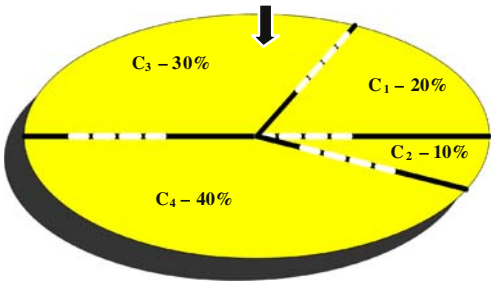


Fig. 5 Roulette

To avoid falling into local optima, we applied an entropy driven mechanism [20] in deciding the mutation rate. The rationale for this mechanism is that we do not want the population to be too diverse otherwise it is hard to approximate an optimum. On the other hand, we do not want it to converge too quickly since it may end up with a local optimum. The meaning of entropy is "lack of information." To measure the diversity level of a generation, we linearly divide chromosomes in that generation into l classes according to their fitness values. This is similar to the concept of histograms in which data points are clustered into l different histogram bins based on their values. The definition of entropy is:

$$-\sum_{i=1}^{l} p_i * log_2(p_i) \tag{3}$$

Where p_i is the probability a chromosome falls into the i^{th} class. This definition shows that the higher the entropy value, the more diverse that population is. This value is then used to dynamically determine the mutation rate. If the population is too diverse, we decrease the mutation rate. Otherwise, we increase it.

At the end of the process, the chromosome with the highest fitness value through all populations is selected as a feasible solution. This chromosome is then decoded to obtain the centroids of clusters as the final output.

It is worth mentioning that the time complexity of clustering is very high. The most time consuming part is to calculate the fitness value with a complexity of $O(n^6)$. In our experiment with 50 chromosomes in each generation and 100 genes in each

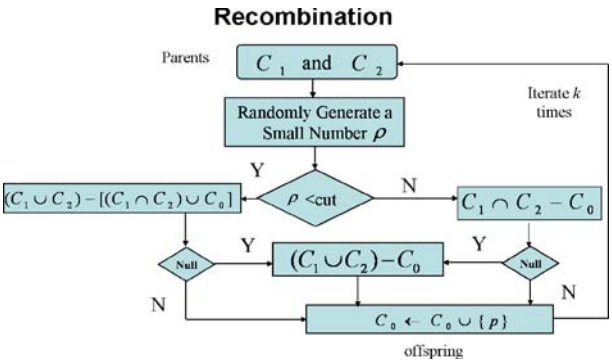


Fig. 6 Recombination algorithm flow chart

chromosome, it takes 1 h and 59 min to run 100 generations. Therefore, the image region clustering is done offline in the preprocessing module and will be re-activated only when the amount of new images being added to the database is larger than a pre-defined threshold.

5 Learning and retrieval framework

5.1 User interest analysis

In this study, we assume that user is only interested in one semantic region of the query image. The goal is to retrieve those images that contain similar semantic regions. In the initial query, user identifies a query image. At this point, no information is provided as to which specific part of the image is of the user's interest. Therefore, for all the images in the image database, we compute the Euclidean distances between the regions of these images and the regions of the query image. The distance between the query image and an image in the image database is represented by the smallest pair-wise distance of their regions. These "representative" distances are then sorted in an ascending order and the top 30 images are returned to the user for feedback.

The user identifies a returned image as "positive" if it is of his/her interest; otherwise the user labels it "negative." With this information at hand, our next step is to estimate the user's interest i.e. which specific region of the query image that user is interested in. Diverse Density (DD) algorithm is applied to accomplish this goal. Diverse Density was first proposed by Maron and Lozano-Pĺęrez in their framework of Multiple Instance Learning [18].

In Multiple Instance Learning (MIL), the label of an individual instance (object) is unknown. Only the label of a set of instances is available, which is called the label of a bag. MIL needs to map an instance to its label according to the information learned from the bag labels. In Content-based Image Retrieval, we have two types of labels—Positive and Negative. Each image is considered a bag of semantic regions (instances). When users supply feedback to the retrieved images, the label of each retrieved image, i.e. bag label, is available. However, the label of each semantic region in that image bag is still unknown because the user only gives feedback to an image as a whole, not to individual semantic regions in that image. The goal of MIL is to estimate the labels (similarity scores) of the test image regions/instances based on the learned information from the labeled images/bags in the training set.

With Diverse Density approach, an objective function called DD function is defined to measure the co-occurrence of similar instances from different bags (images) with the same label. The target of DD is to find a point which is the closest to all the positive images and farthest from all the negative images. The framework of DD [18] is briefly explained below.

We denote the positive bags as $B_1^+, B_2^+, ..., B_n^+$ and the negative bags as $B_1^-, B_2^-, ..., B_m^-$. The j^{th} instance of bag B_i^+ is represented as B_{ij}^+, while the j^{th} instance of bag B_i^- is written as B_{ij}^-. Each bag may contain any number of instances, but every instance must be represented by a k dimensional vector where k is a constant.

Different semantic concepts may share some common low-level features, but not all k dimensions contribute equally. Therefore, given a semantic concept, greater

weights shall be assigned to features with more distinguishing power. For example, color and texture features shall be assigned greater weights for "grass" compared to shape features. We denote this weight vector $\alpha = \{\alpha_1, \alpha_2, ..., \alpha_k\}$.

For any point $p = \{p_1, p_2, ..., p_k\}$ in the feature space, the Diverse Density is defined by the probability of it being our target point, given all the positive and negative bags. So the point we are looking for is the one that maximize the probability below.

$$Argmax\, P_r(p|B_1^+, B_2^+, ..., B_n^+, B_1^-, B_2^-, ..., B_m^-) \tag{4}$$

Assuming a uniform prior over the concept location $P_r(p)$ and conditional independence of the bags given the target concept p, the above function equals to

$$Argmax \prod_i P_r(p|B_i^+) \prod_i P_r(p|B_i^-) \tag{5}$$

The following model was introduced by Maron and Lozano-PÍęrez for estimating hypothesis $h = \{p_1, p_2, ..., p_k, \alpha_1, \alpha_2, ..., \alpha_k\}$.

$$P_r(B_{ij} = p) = exp\left(-\sum_1^k \alpha_q (B_{ijq} - p_q)^2\right) \tag{6}$$

The goal is to find such a hypothesis h such that the above function reaches its maximum. We apply EM (Expectation-Maximization) algorithm as proposed in [31]. EM starts with an initial hypothesis h, and then repeatedly performs E-step and M-step. In E-step, the current hypothesis h is used to pick one instance from each bag which is most likely to be the one responsible for the label of the bag. In M-step, a two-step gradient ascent search (Quasi-Newton search) is performed on DD algorithm to find a new h' that maximizes the above function. In the new iteration h' replaces h. The loop continues until the algorithm converges.

We then use the final result h, the point of the user's interest, to find the cluster this point h belongs to. Hence all the other image regions in this cluster can be located. However, we cannot simply reduce the search space to this cluster alone because it is not a rare case that a particular region is closer to some regions in another cluster than some regions within the same cluster. This situation is illustrated in Fig. 7. Suppose the query image region is a "horse" region in Image B. It is

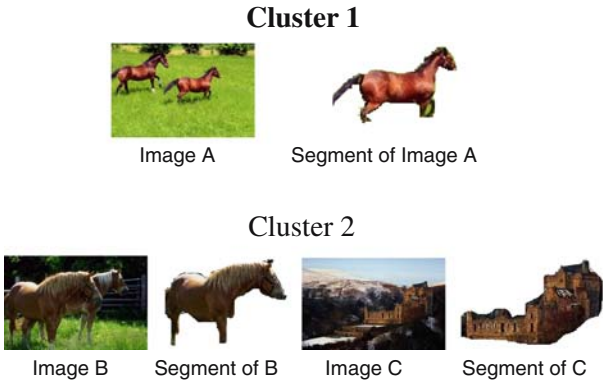


Fig. 7 Image region clusters

in the same cluster with the "castle" region of Image C because these two regions share similar low level features. However, Image B is conceptually closer to Image A whose "horse" region is in another cluster. Therefore, in our system, we choose three clusters whose centroids are the closest to the query region. As an image is composed of several semantic regions, it can fall into any cluster that has at least one of its semantic regions. We then group all the images that have at least one semantic region fall into the three clusters mentioned above and take it as the reduced search space for a given query region. The effectiveness of this reduction is presented in Section 6.

5.2 Learning and retrieval

As mentioned above, DD algorithm is applied to analyze the user's interest after the initial query. Yet due to the large amount of noise in the image data set, we cannot guarantee that the user's interest is exactly h. Instead, it is taken as our initial hypothesis and the system continues interacting with user to collect more feedbacks. The output of DD is a group of instances, one from each image, that contribute most to the image (bag) label. Specifically, in the output of DD, instances that come from positive bags are positive instances. Because of these instances, their corresponding bags are labeled positive. We then construct the training sample set according to this output of DD. This is then fed into One-Class Support Vector Machine as the initial training sample set, which further learns and models user's interest and refines the retrieval result in the following iterations.

One-Class classification is a kind of unsupervised learning mechanism. It tries to assess whether a test point is likely to belong to the distribution underlying the training data. In our case, the training set is composed of positive samples only. Figure 8 shows how positive image regions are all alike and should be in one class while it is inappropriate to group negative image regions into a single class. In Fig. 8, image regions are outlined by white lines. Suppose the user's interest is an "eagle" object, then ideally positive image regions shall be those "eagle" regions. However, negative image regions can be anything other than "eagle". As shown in Fig. 8, negative image regions can be "flower", "fish", "glass", etc.

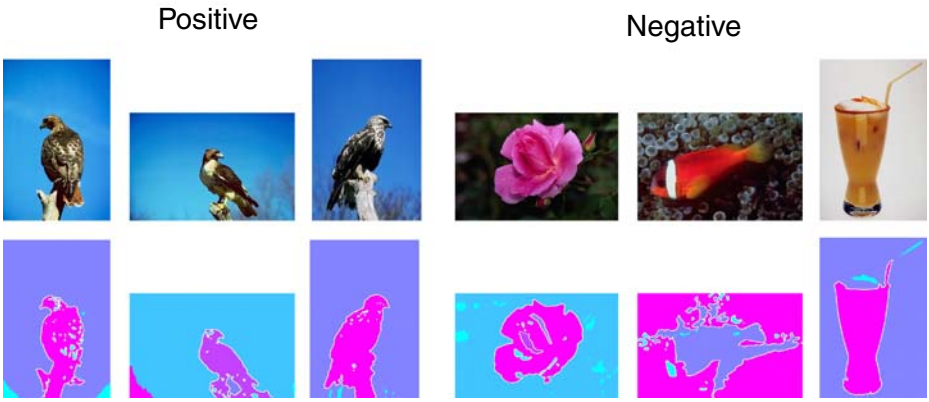


Fig. 8 One class classification

One-Class SVM has so far been studied in the context of SVMs [22]. The objective is to create a binary-valued function that is positive in those regions of input space where the data predominantly lies and negative elsewhere. The idea is to model the dense region as a "ball." In our case, positive instances are inside the "ball" and negative instances are outside. If the origin of the "ball" is **o** and the radius is r, a point $\mathbf{p_i}$, in this case an instance (image region) represented by an 19-dimension feature vector, is inside the "ball" *iff* $\|\mathbf{p_i} - \mathbf{o}\| \leqslant r$. This "ball" is actually a hyper-sphere. The goal is to keep this hyper-sphere as "pure" as possible and include most of the positive objects. Since this involves a non-linear distribution in the original space, the strategy of Schölkopf's One-Class SVM is first to do a mapping θ to transform the data into a feature space F corresponding to the kernel K:

$$\theta(p_1) \cdot \theta(p_2) \equiv K(p_1, p_2) \tag{7}$$

where p_1 and p_2 are two data points. In this study, we choose to use Radial Basis Function (RBF) Machine below.

$$K(p_1, p_2) = exp(\|p_1 - p_2\|/2\sigma) \tag{8}$$

Mathematically, One-Class SVM solves the following quadratic problem:

$$\min_{\omega,\zeta,\rho} \frac{1}{2}\|\omega\| - u\rho + \frac{1}{n}\sum_{i=1}^{n} \zeta_i \tag{9}$$

subject to

$$(\omega \cdot \theta(p_i)) \geqslant \rho - \zeta_i, \ \zeta_i \geqslant 0 \text{ and } i = 1, ...n \tag{10}$$

where ζ_i is the slack variable, and $u \in (0, 1)$ is a parameter that controls the trade off between maximizing the distance from the origin and containing most of the data in the region created by the hyper-sphere. It actually corresponds to the ratio of "outliers" in the training dataset. If ρ and ω are a solution to this problem, then the decision function is $f(x) = sign(\omega \cdot \theta(p) - \rho)$ and it will be 1 for most examples

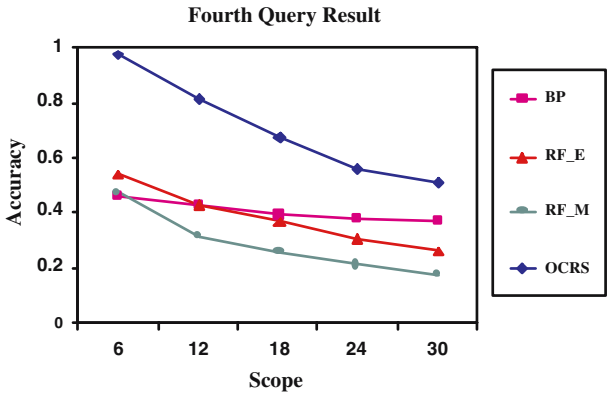


Fig. 9 Retrieval accuracy after the fourth query

p_i contained in the training set. Some images may actually contain more than one positive region. Therefore, we cannot assume that only one region in each image is positive. Suppose the number of positive images is n and the number of all semantic regions in the training set is N. Then the ratio of "outliers" in the training set is set to:

$$u = 1 - \left(\frac{n}{N} + z\right) \tag{11}$$

z is a small number used to adjust the u in order to alleviate the above mentioned problem. Our experiment results show that $z = 0.01$ is a reasonable value.

The training set as well as the parameter u are fed into One-Class SVM to obtain ρ and ω, which are used to calculate the value of the decision function for the test data, i.e. all the image regions in the database. Each image region will be assigned a "score" by $\omega \cdot \theta(p) - \rho$ in the decision function. The higher the score, the more likely this region is in the positive class. The similarity score of each image is then set to the highest score of all its regions.

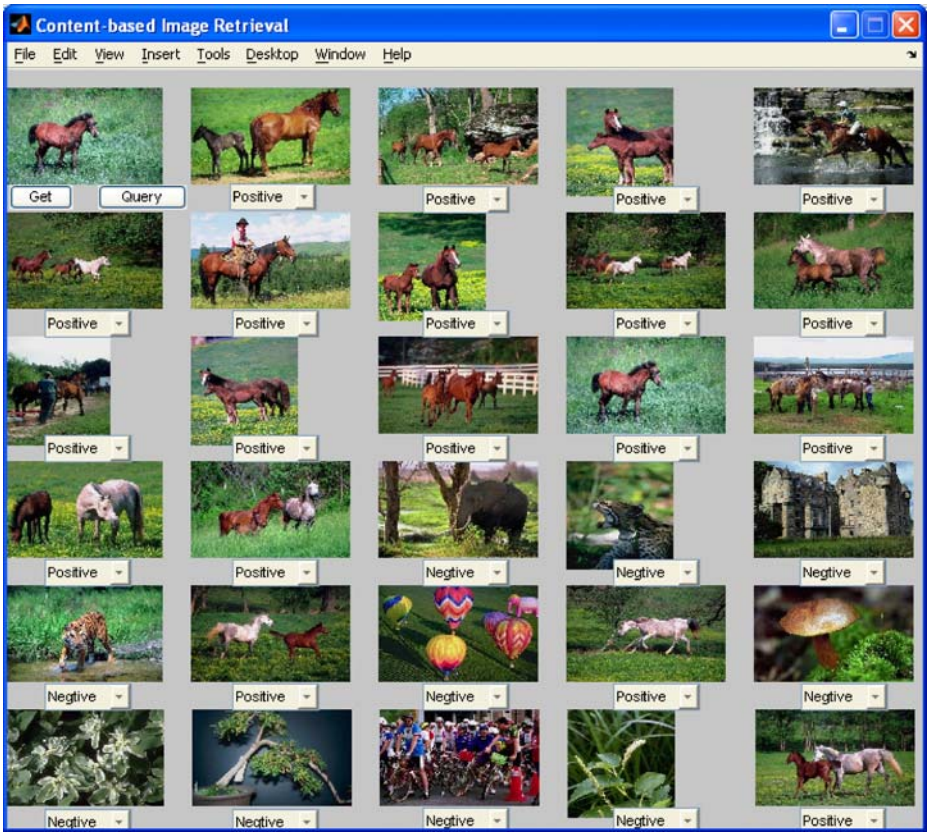


Fig. 10 Query results of OCRS after the fourth iteration

6 System performance evaluation

The experiment is conducted on a Corel image database consisting of 10,000 images from 100 categories. After segmentation, there are in total 49,584 image segments. We tested the system performance under different clustering schemes by dividing the entire set of image regions into 40 to 150 clusters. Each time we increase the number of clusters by 10 and find that when the number of clusters k=100, the result is most reasonable in terms of the balance between accuracy and reduction of search space. After the initial query, according to the hypothesis generated by DD, we pull out the three closest clusters as the reduced search space. All the images that have at least one segment fall into these three clusters are identified and fed into the learned One-Class SVM for classification. Sixty images are randomly chosen from 20 categories as the query images. According to our experiment, the search space, in terms of the number of images in the three candidate clusters, is reduced to 13.4% of the original search space (10,000 images) on average.

We compare the performance of our system with two other relevance feedback algorithms: (1) Neural Network based Multiple Instance Learning (MIL) algorithm with relevance feedback [12]; (2) General feature re-weighting algorithm [21] with relevance feedback. For the latter, both Euclidean and Manhattan distances are tested.

Five rounds of relevance feedback are performed for each query image - Initial (no feedback), First, Second, Third, and Fourth. The accuracy rates with different scopes, i.e. the percentage of positive images within the top 6, 12, 18, 24 and 30 retrieved images, are calculated. Figure 9 shows the result after the Fourth Query. "BP" is the Neural Network based MIL which uses both positive and negative examples. "RF_E" is the feature re-weighting method using Euclidean Distance while "RF_M" uses Manhattan Distance. "OCRS" is the proposed system.

It can be gleaned from Fig. 9 that while the search space is substantially reduced, the accuracy of the proposed framework still outperforms all the other three algorithms. It also can be seen that the Neural Network based MIL (BP) shows a better result than that of general feature re-weighting algorithm after four rounds of learning. In addition, the performance of RF_E using Euclidean Distance is slightly better than that of RF_M which uses Manhattan Distance. Figure 10 shows the fourth

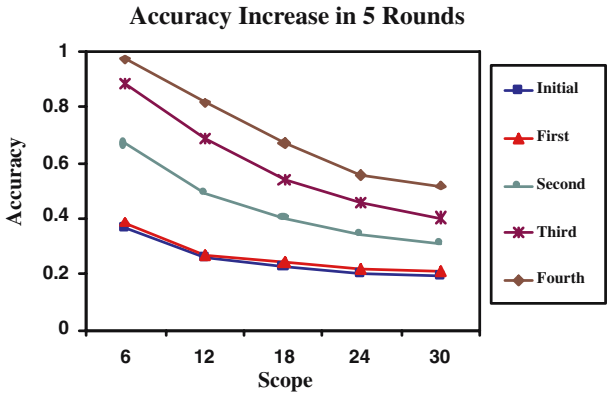


Fig. 11 Retrieval results of OCRS across five iterations

query results of "OCRS", given the query image on the upper left corner of the interface. In this example, "horse" is the user's interest. It can be seen that there are 20 "horse" images out of the top 30 returned images.

It is worth mentioning that, the number of positive images increases steadily through each iteration. Figure 11 gives a concrete view as to how accuracy rates of our algorithm increases across five iterations.

7 Conclusion

In this paper, we proposed a framework, OCRS, for single region based image retrieval. OCRS strives to solve two crucial problems in this area, i.e. time complexity due to the huge amount of high-dimensionality data; semantic gap between low level features and human subjectivity. Specifically in preprocessing, a Genetic Algorithm based clustering mechanism is proposed to reduce the search space. An efficient image segmentation algorithm—WavSeg is implemented to divide an image into semantic regions. We then adopt Diverse Density to do the initial analysis of user's interest. As initial hypothesis, the output of DD is fed into One-Class SVM in the image retrieval phase. The advantage of our algorithm is that it targets image region retrieval instead of the whole image, which is more reasonable since the user is often interested in only one region in the image. The proposed work also transforms the One-Class SVM learning for region-based image retrieval into a Multiple Instance Learning problem. In addition, due to the robustness of Genetic Algorithm in approximating global optima and the generality of One-Class SVM, the proposed system has proved to be effective in better identifying the user's real need and removing the noise data.

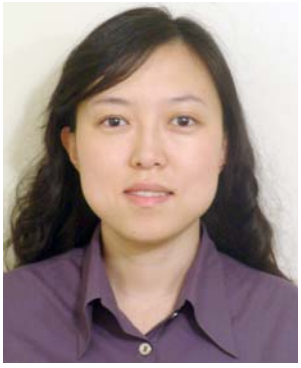
Acknowledgements The work of Chengcui Zhang was supported in part by SBE-0245090 and the UAB ADVANCE program of the Office for the Advancement of Women in Science and Engineering and the UAB Faculty Development Award.

References

1. Andrews S, Tsochantaridis I, Hofmann T (2003) Support vector machines for multiple-instance learning. In: Advances in neural information processing systems 15, pp 561–568. MIT Press, Cambridge, MA
2. Carson C, Belongie S, Greenspan H, Malik J (2002) Blobworld: image segmentation using expectation-maximization and its application to image querying. IEEE Trans Pattern Anal Mach Intell 24(8)
3. Chen S-C, Rubin SH, Shyu M-L, Zhang C (2006) A dynamic user concept pattern learning framework for content-based image retrieval. IEEE Trans Syst Man Cybern Part C 36(6): 772–783
4. Chen S-C, Sista S, Shyu M-L, Kashyap RL (2000) An indexing and searching structure for multimedia database systems. In: Proceedings of the IS&T/SPIE conference on storage and retrieval for media databases, pp 262–270
5. Chen YX, Wang JZ (2004) Image categorization by learning and reasoning with regions. J Mach Learn Res 5:913–939
6. Chen YX, Wang J, Krovetz R (2005) CLUE: cluster-based retrieval of images by unsupervised learning. IEEE Trans Image Process 14(8):1187–1201
7. Chen Y, Zhou X, Tomas S, Huang TS (2001) One-class SVM for learning in image retrieval. In: Proceedings of IEEE international conference on image processing
8. Dietterich TG, Lathrop RH, Lozano-Perez T (1997) Solving the multiple-instance problem with axis-parallel rectangles. Artif Intel J 89:31–71

9. Ferhatosmanoglu H, Stanoi I, Agrawal D, Abbadi AE (2001) Constrained nearest neighbor queries. In: Proceedings of SSTD
10. Gondra I, Heisterkamp DR (2004) Adaptive and efficient image retrieval with one-class support vector machines for inter-query learning. WSEAS Trans Circuits Syst 3(2):324–329, April
11. Holland JH (1975) Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor
12. Huang X, Chen S-C, Shyu M-L, Zhang C (2002) User concept pattern discovery using relevance feedback and multiple instance learning for content-based image retrieval. In: Proceedings of the 3rd international workshop on Multimedia Data Mining (MDM/KDD'2002), pp 100–108
13. Jing F, Li MM, Zhang L, Zhang H-J, Zhang Bo (2003) Learning in region-based image retrieval. In: Proceedings of international conference on image and video retrieval
14. Jing F, Li MM, Zhang L, Zhang H-J, Zhang Bo (2003) Support vector machines for region-based image retrieval. In: Proceedings of IEEE international conference on multimedia & expo
15. Kanungo T, Mount D, Netanyahu N, Piatko CD, Silverman R, Wu AY (2002) An efficient K-means clustering algorithm: analysis and implementation. IEEE Trans Pattern Anal Mach Intell 24(7), July
16. Kim DH, Chung CW (2003) Qcluster: relevance feedback using adaptive clustering for content based image retreival. In: Proceedings of SIGMOD conference
17. Krishnamachari S, Abdel-Mottaleb M (1999) Hierarchial clustering algorithm for fast image retrieval. In: Proceedings of IS&T/SPIE conference on storage and retreival for image and video databases VII. San Jose, California, pp 427–435, Jan
18. Maron O, Lozano-Perez T (1998) A framework for multiple instance learning. Advances in natural information processing system 10. MIT Press, Cambridge, MA
19. Ramon J, De Raedt L (2000) Multi-instance neural networks. In: Proceedings of the ICML 2000 workshop on attribute-value and relational learning
20. Rosca J (1995) Entropy-driven adaptive representation. In: Proceedings of the workshop on genetic programming: From theory to real-world applications, pp 23–32
21. Rui Y, Huang TS, Mehrotra S (1997) Content-based image retrieval with relevance feedback in MARS. In: Proceedings of the international conf. on image processing, pp 815–818
22. Schölkopf B, Platt JC et al (1999) Estimating the support of a high-dimensional distribution. Microsoft research corporation technical report MSR-TR-99-87
23. Su Z, Zhang HJ, Li S, Ma SP (2003) Relevance feedback in content-based image retrieval: bayesian framework, feature subspaces, and progressing learning. IEEE Trans Image Process 12(8):924–937
24. Vladimir EC, Murray AT (1997) Spatial clustering for data mining with genetic algorithms. Technical report FIT-TR-97-10, Queensland University of Technology, Faculty of Information Management, September
25. Wang L, Liu L, Khan L (2004) Automatic image annotation and retrieval using subspace clustering algorithm. In: Proceedings of the second ACM international workshop on multimedia databases, pp 100–108
26. Wang J, Zucker J-D (2000) Solving the multiple instance learning problem: a lazy learning approach. In: Proceedings of the 17th international conference on machine learning, pp 1119–1125
27. Wu P, Manjunath BS (2001) Adaptive nearest neighbor search for relevance feedback in large image databases. In: Proceedings of ACM multimedia
28. Yang C, Dong M, Fotouhi F (2005) Region based image annotation through multiple-instance learning. In: Proceedings of ACM international conference on multimedia, Singapore, 6–11 Nov
29. Yang C, Lozano-Prez T (2000) Image database retrieval with multiple-instance learning techniques. In: Proceedings of the 16th international conference on data engineering, pp 233–243
30. Zhang C, Chen S-C, Shyu M-L, Peeta S (2003) Adaptive background learning for vehicle detection and spatio-temporal tracking. In: Proceedings of the 4th IEEE Pacific-rim conference on multimedia, pp 1–5
31. Zhang Q, Goldman SA (2001) EM-DD: An improved multiple-instance learning technique. Adv Neural Inf Process Syst (NIPS) pp 14
32. Zucker J-D, Chevaleyre Y (2001) Solving multiple-instance and multiple-part learning problems with decision trees and decision rules. Application to the mutagenesis problem. In: Proceedings of the 14the Biennial Conference of the Canadian Society for Computational Studies of Intelligence, AI 2001, pp 204–214

Chengcui Zhang is an Assistant Professor of Computer and Information Science at University of Alabama at Birmingham (UAB) since August, 2004. She received her Ph.D. from the School of Computer Science at Florida International University, Miami, FL, USA in August, 2004. She also received her bachelor and master degrees in Computer Science from Zhejiang University in China. Her research interests include multimedia databases, multimedia data mining, image and video database retrieval, bioinformatics, and GIS data filtering. She is the recipient of several awards, including the IBM Unstructured Information Management Architecture (UIMA) Innovation Award, UAB ADVANCE Junior Faculty Research Award from the National Science Foundation, UAB Faculty Development Award, and the Presidential Fellowship and the Best Graduate Student Research Award at FIU.



Xin Chen received her Master's degree in Computer Science from University of Science and Technology Beijing, China, in 2002. From 2004 to present, she has been pursuing her Ph.D. degree in the Computer and Information sciences Department in University of Alabama at Birmingham. Her research interests include Content-based Image Retrieval, multimedia data mining, and spatio–temporal data mining.