Locating Regions of Interest in CBIR with Multi-Instance Learning Techniques

Zhi-Hua Zhou, Xiao-Bing Xue, and Yuan Jiang

National Laboratory for Novel Software Technology
Nanjing University, Nanjing 210093, China
{zhouzh,xuexb,jiangy}@lamda.nju.edu.cn

Abstract. In content-based image retrieval (CBIR), the user usually poses several labelled images and then the system attempts to retrieve all the images relevant to the target concept defined by these labelled images. It may be helpful if the system can return relevant images where the *regions of interest* (ROI) are explicitly located. In this paper, this task is accomplished with the help of multi-instance learning techniques. In detail, this paper proposes the CkNN-ROI algorithm, which regards each image as a *bag* comprising many instances and picks from positive bag the instance that has great chance to meet the target concept to help locate ROI. Experiments show that the proposed algorithm can efficiently locate ROI in CBIR process.

1 Introduction

With the rapid increase of the volume of digital image collections, content-based image retrieval (CBIR) has attracted a lot of research interests in recent years [5]. Here the main difficulty lies in the gap between the high-level image semantics and the low-level image features, due to the rich content but subjective semantics of an image. Although much endeavor has been devoted to bridging this gap [4][11], it remains an unsolved problem at present. Nevertheless, many good CBIR systems have already been developed.

In CBIR, the user usually poses several labelled images, i.e. positive or negative images corresponding to a target concept, in the query and relevant feedback process, and the CBIR system attempts to retrieve all the images relevant to the target concept from the image database. It is noteworthy that although the user feeds whole images to the system, usually he or she is only interested in some regions in these images, i.e. *regions of interest* (ROI). In applications involving the scan of huge volume of images to detect suspect areas, such as to some medical or military purposes, it may be valuable if the ROI can be identified and exhibited when the retrieved images are presented to the user. Even in common CBIR scenarios, considering that the system usually returns a lot of images, explicitly showing the ROI may be helpful because it could help the user recognize the images he or she really wants more quickly. Unfortunately, at present few CBIR systems can return retrieved images where the ROI has been located.

In this paper, the CkNN-ROI algorithm is proposed to address the problem of locating ROI in CBIR. CkNN-ROI regards each image as a *bag* comprising many instances. The labelled positive or negative images provided by the user are regarded respectively as positive or negative training bags. For every image in the database, CkNN-ROI determines whether or not it is a positive bag with the help of some adjacent training bags, and if the image is a positive bag then the instance which has the biggest chance to meet the target concept is picked out while its corresponding image region is regarded as ROI. Experiments show that this algorithm can locate ROI in the CBIR process with high efficiency.

The rest of the paper is organized as follows. Section 2 briefly introduces multi-instance learning and its application in CBIR. Section 3 proposes the CkNN-ROI algorithm. Section 4 reports on the experiments. Finally, Section 5 concludes and raises several issues for future work.

2 Multi-Instance Learning and CBIR

In *multi-instance learning* [10], the training set is composed of many *bags* each contains many instances. A bag is positively labelled if it contains at least one positive instance and negatively labelled otherwise. The task is to learn something from the training bags for correctly labelling unseen bags. This task is difficult because unlike supervised learning where all the training instances are labelled, here the labels of the individual instances are unknown. It is obvious that if a whole image is regarded as a bag while its regions are regarded as instances, then the problem of determining whether an image is relevant to a target concept can be viewed as a multi-instance problem. Therefore, multi-instance learning has great potential in tasks involving image analysis.

Maron and Ratan [3] applied multi-instance learning techniques to natural scene classification. In their work each image was initially smoothed using a Gaussian filter and subsampled to an 8×8 matrix of *color blobs* where each blob was a 2×2 set of pixels within the 8×8 matrix. Then, different *bag generators* were used to transform various configurations of blobs of each image, such as *rows*, *single blob with neighbors* (SBN), *two blobs with no neighbors*, etc., into instances of the corresponding bag. Finally, the Diverse Density algorithm [2] was used to learn the target concept. Although natural scene classification is different from CBIR, this work gave some illumination on the application of multi-instance learning techniques to CBIR.

Yang and Lozano-Pérez [7] first applied multi-instance learning techniques to CBIR. They transformed color images into gray-scale images at first. Then, they divided each image into many overlapping regions. Regions with low variances were thrown out and each of the remaining ones was smoothed and subsampled to a low-resolution $h \times h$ matrix. Through concatenating these elements, an h^2-dimensional feature vector was generated. After subtracting the mean of the feature vector from it and then dividing it by its standard deviation, a new h^2-dimensional feature vector was obtained, which was used to describe the corresponding instance. Finally, a variation of the weighted correlation statistic was utilized to measure the similarity between feature vectors, and the Diverse Density algorithm was employed to learn the target concept. It is worth noting that this bag generator requires converting color images into gray-scale images, therefore it is not suitable to the process of color images.

Zhang et al. [9] enhanced Maron and Ratan's SBN bag generator [3] through exploiting a variety of image processing techniques such as wavelet transformation. They used this bag generator to convert the images to bags and employed the EM-DD algorithm [8], a variant of the Diverse Density algorithm which utilizes the EM technique, to learn the target concept. It is noteworthy that besides boolean labels, they also tried real-valued labels for each bag indicating how well the corresponding image meets the target concept.

Zhou et al. [13] developed the bag generator ImaBag, which was derived from an SOM-based image segmentation technique. Here the pixels in each image were clustered based on their color and spatial features by an SOM neural network, and then the clustered blocks were merged into a specific number of regions. Each region was represented with a feature vector formed by its mean R, G, B values, which was regarded as an instance in the bag corresponding to the image. Then the Diverse Density algorithm was used to learn the target concept.

3 C*k*NN-ROI

It can be seen from Section 2 that among current multi-instance learning algorithms, the Diverse Density algorithm and its variants have already been used in CBIR. This is not strange because Diverse Density is the first practical multi-instance learning algorithm, which employs gradient ascent with multiple starting points to search for the point with the maximum *diverse density* in the feature space [2]. The *diverse density* at a point is defined to be a measure of how many different positive bags have instances near that point, and how far the negative instances are from that point. It is obvious that the Diverse Density algorithm can be used to locate ROI because after training process the point with maximum diverse density in the feature space is identified, therefor for a positive image bag the instance closest to that point should be the one corresponding to ROI. In fact, this property has already been exploited in Maron and Ratan's work on natural scene classification [3].

However, the Diverse Density algorithm suffers from huge time cost. In order to get the concept point, i.e. the point with the maximum diverse density in the feature space, this algorithm has to perform gradient-based optimizations starting from every instance in every positive bags. Suppose the training set contains m positive bags and each positive bag contains l instances. Then the total number of gradient-based optimizations required by the Diverse Density algorithm is $(m \times l)$. In natural scene classification such a time cost is tolerable because all the classes of the natural scenes are known when the the image database is constructed. Therefore, the concept points for different classes can be pre-computed and a new image can be quickly classified through computing its distances to different concept points. However, in CBIR the classes that may occur are not known when the image database is constructed, because the possible target concepts the user may query for may be infinite. Thus, the concept point can only be computed after the user provides labelled images. Since the gradient-based optimizations have to be executed online, the retrieval process will be time-consuming, which might not be acceptable because usually the user wants to get the retrieval results as soon as possible.

Based on the above recognition, nearest neighbor algorithms [1] appear to be better choices because they only involve the computation of distances while the distances between training examples can be pre-computed. Actually, there is a nearest neighbor style multi-instance learning algorithm, i.e. Citation-kNN [6]. This algorithm borrows the notion of *citation* and *reference* of scientific literatures in the way that a bag is labelled through analyzing not only its neighboring bags but also the bags that regard the concerned bag as a neighbor. In computing the distances between different bags, the *minimal Hausdorff distance* is employed, as shown in Eq. 1 where A and B are two different bags, i.e. $A = \{a_1, a_2, \cdots, a_m\}$ and $B = \{b_1, b_2, \cdots, b_h\}$ while a_i $(1 \leq i \leq m)$ and b_j $(1 \leq j \leq h)$ are the instances.

$$\mathrm{Dist}\,(A, B) = \underset{\substack{1 \leq i \leq m \\ 1 \leq j \leq h}}{\mathrm{MIN}} \left(\mathrm{Dist}\,(a_i, b_j)\right) = \underset{a \in A}{\mathrm{MIN}}\, \underset{b \in B}{\mathrm{MIN}} \|a - b\| \tag{1}$$

The Citation-kNN algorithm achieved the best performance on the *Musk* data, a popular benchmark test for multi-instance learning algorithms, at the time it was proposed [6]. Recently, Zhou et al. [12] developed a variant of Citation-kNN for a web mining task through modifying the minimum Hausdorff distance for text features and obtained success. This demonstrates the great application potential of the k-NN style multi-instance learning algorithms. However, up to now such kind of algorithms have not been applied to tasks involving image analysis, while the CkNN-ROI algorithm described below is the first attempt to introduce them into CBIR.

In fact, the Citation-kNN algorithm can be directly applied to CBIR without any modification. However, this algorithm can not locate ROI by itself. Therefore, the CkNN-ROI algorithm is proposed, which works in two steps.

In the first step, the label of a new bag is determined in the same way as that of Citation-kNN. That is, for a given new bag, its r-nearest neighboring training bags (r-*references*) as well as the training bags (c-*citers*) which regard the new bag as its c-nearest neighbor are identified according to Eq. 1, and then the label of the new bag is determined by voting among these training bags.

In the second step, if the new bag is deemed as negative then nothing will be done because a negative image should contain no ROI corresponding to the target concept; otherwise the following process is executed. For each instance in the new bag, its k-nearest neighboring training bags can be identified according to Eq. 1 through regarding the concerned instance as a bag containing only one instance. Each of these training bags contributes a score to the estimation of the chance for the instance to meet the target concept. Intuitively, the closer the positive bags while the farther the negative bags, the bigger the chance.

Table 1. Pseudo-code describing the CkNN-ROI algorithm

Algorithm:	CkNN-ROI
Input:	New bag B^*, training set T, parameters to set: r, c, k
Output:	Instance of B^* corresponding to ROI
Process:	Labelling Step: 1) Find B^*'s r-nearest bags in T according to Eq. 1. Add them to R. $p \leftarrow$ the number of positive bags in R $n \leftarrow$ the number of negative bags in R 2) Find all bags in T whose c-nearest bags contain B^* according to Eq. 1. Add them to C. $p \leftarrow p +$ the number of positive bags in C $n \leftarrow n +$ the number of negative bags in C 3) If $p > n$ then goto Locating Step; otherwise B^* is negative which contains no ROI. Exit. Locating Step: 1) For each instance in B^*, find its k-nearest bags in T according to Eq. 1 through regarding the instance itself as a bag. Add them to Ω. Compute *Score* according to Eq. 2. 2) Find the instance, say I^*, having the biggest *Score*. 3) Return I^*.

Therefore, an exponential distance is employed and the equation for computing the score is empirically developed as shown in Eq. 2, where B^* is the new bag while I_v^* is its vth instance, $\Omega = \{k\text{-nearest neighboring training bags of } I_v^*\}$, I_{ij} is an instance of B_i, $class(B_i)$ returns $+1$ if B_i is positive and -1 otherwise, $||\cdots||_2$ is the l_2-norm.

$$Score_v = \sum_{i:B_i \in \Omega} (-1)^{class(B_i)} \exp\left(-\left(\min_{I_{ij} \in B_i} \|I_v^* - I_{ij}\|_2\right)^2\right) \quad (2)$$

The instance with the biggest score is deemed as a positive instance corresponding to the target concept, while its corresponding image region is regarded as the ROI. The pseudo-code of the CkNN-ROI algorithm is shown in Table 1.

Note that the first step of the CkNN-ROI algorithm reassembles the Citation-kNN algorithm, which has computed the distances between all the instances. Therefore, the computational cost of the CkNN-ROI algorithm is only slightly bigger than that of the Citation-kNN algorithm, where the additional cost lies in directly putting the existing $\min_{I_{ij} \in B_i} \|I_v^* - I_{ij}\|_2$ values into Eq. 2 and then picking out the instance with the biggest *Score*. Moreover, the distances between all the different training bags can be pre-computed when the image database is constructed. Therefore, comparing to Diverse Density, CkNN-ROI can be much more efficient, which is desirable in CBIR because the user can get the retrieval results much more quickly.

4 Experiments

An image database consisting of 500 COREL images is used in the experiments, which includes 100 images from each of the five image types: *castle*, *firework*, *mountain*, *sunset*, and *waterfall*. Here each image type corresponds to a target concept to be retrieved. A training set comprising 50 images is created by randomly choosing 10 images from each of the five image types. The remaining 450 images constitute a test set. The training-test partition is randomly generated for ten times where the generated training sets are not overlapped, and the average statistics is recorded.

The CkNN-ROI algorithm is compared with Diverse Density. For CkNN-ROI, the parameters r and c are set to 3 and 5, respectively, which is the recommended parameter setting of Citation-kNN [6]; the parameter k is set to 3, which has not been finely tuned. For Diverse Density, the *single point scaling* concept is adopted [2]. Both algorithms are facilitated with the same bag generator, i.e. the SBN presented by Maron and Ratan [3]. Images are first filtered and subsampled to 8×8. Then an SBN is defined as the combination of a single blob with its four neighboring blobs (up, down, left, right). The sub-image is described as a 15-dimensional vector, where the first three features represent the mean R, G, B values of the central blob and the remaining twelve features correspond to the differences in mean color values between the central blob and other four neighboring blobs respectively. Therefore, each image bag is represented by a collection of thirty-six 15-dimensional feature vectors obtained by using each of the blobs not along the border as the central blob. Note that in order to locate ROI more precisely, here 1×1 blob is used instead of 2×2 blob used in [3]. Fig. 1 illustrates the bag generator.

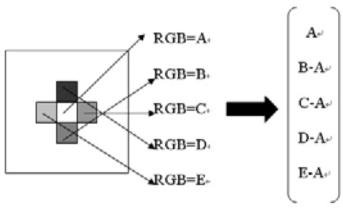


Fig. 1. The SBN bag generator

The experiments are performed on a machine with Intel Pentium III 533Hz CPU, 384MB memory, and Windows XP operating system. The average time cost per query is compared, as shown in Table 2.

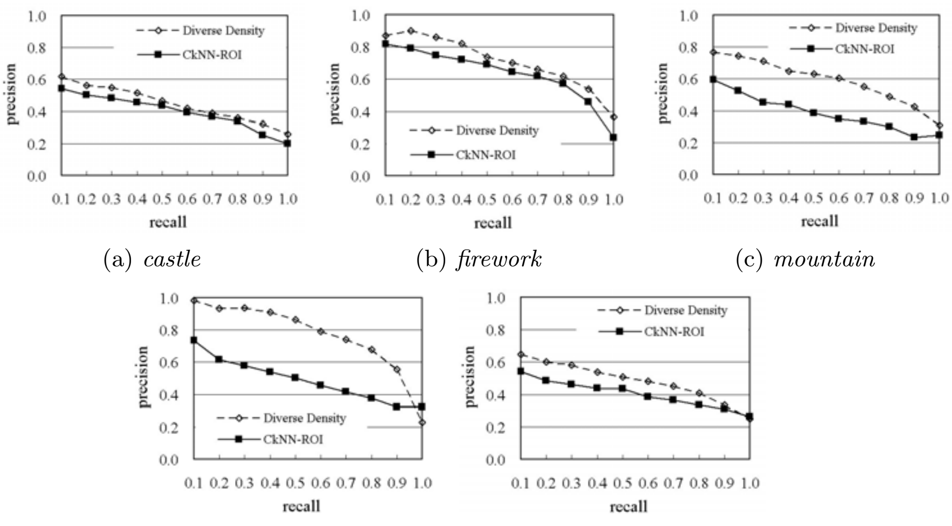
Table 2 shows that the CkNN-ROI algorithm is about 45,234 ($= 14,475/0.32$) times faster than the Diverse Density algorithm in dealing with CBIR queries. The process time of CkNN-ROI, i.e. 0.32 second per query, is a very good record even in the context of state-of-the-art CBIR systems. Note that if the image

Table 2. Comparison on the average time cost per query

	Diverse Density	CkNN-ROI
Time(second)	14,475	0.32

database is larger, the distances between more bags should be pre-computed, but fortunately the online process time of CkNN-ROI will only increase linearly as well as Diverse Density. While even on the relatively small experimental image database, the process time of Diverse Density, i.e. 14,475 seconds per query, is unbearable because it is difficult to anticipate such a patient user who will wait more than 4 hours for a query! This discloses the fact that even though the Diverse Density algorithm could obtain good retrieval results, it could hardly be really used in CBIR dues to its overwhelmingly low efficiency.

Fig. 2 shows the precision-recall graphs of the retrieval performance of CkNN-ROI and Diverse Density on the target concepts. Note that each point shown in the graphs is the average result of 10 queries of the same type. This figure shows that the performance of CkNN-ROI is comparable to that of Diverse Density on three target concepts, i.e. *castle*, *firework*, and *waterfall*; while apparently worse than that of Diverse Density on two target concepts, i.e. *mountain* and *sunset*.



(a) *castle* (b) *firework* (c) *mountain*

(d) *sunset* (e) *waterfall*

Fig. 2. Precision-recall graphs for different target concepts

The ROI locating ability of CkNN-ROI and Diverse Density are also compared. In detail, ROI of each image is manually marked in advance. Then, for

Fig. 3. The ROI located by Diverse Density and CkNN-ROI. Each row shows five pairs of example images on the target concepts *castle*, *firework*, *mountain*, *sunset*, and *waterfall*, respectively. In each pair the first image is obtained with Diverse Density while the second one is obtained with CkNN-ROI.

each relevant image in the database, if the ROI returned by the algorithm is covered by the real ROI then it is recorded as a success. Note that here whether an image is relevant to the target concept or not is determined by its real label, therefore in this way the ROI locating performance and the retrieval performance can be separately evaluated. The ratio of the number of successes against the total number of relevant images on each target type is computed and compared in Table 3. Fig. 3 shows some example images with located ROI.

Table 3. Comparison on the success ratio of ROI locating

Target Concept	Diverse Density	CkNN-ROI
castle	0.668	0.608
firework	0.719	0.484
mountain	0.678	0.712
sunset	0.386	0.074
waterfall	0.767	0.409

Table 3 shows that the ROI locating performance of CkNN-ROI is worse than that of Diverse Density on *firework*, *sunset*, and *waterfall*, while comparable or better on *castle* and *mountain*. This is not difficult to explain because the bag generator used here mainly considers color information, therefore the instances can be viewed as some kind of color patterns. Diverse Density attempts to find a color pattern shared by all the positive bags but no negative bags, and uses this pattern to locate ROI; while CkNN-ROI attempts to evaluate each color pattern through finding its k nearest neighboring bags that have similar color patterns, and the more positive bags among these k bags, the more likely the concerned color pattern is regarded as ROI. So, CkNN-ROI is a local method while Diverse Density is a somewhat global method. If the image semantics is strongly coupled with a specific color pattern, then all the positive bags do share some specific color patterns, and in this case Diverse Density should perform better than CkNN-ROI; while if the image semantics is not strongly coupled with a specific color pattern, then it is difficult to find out a color pattern shared by all the positive bags, and in this case CkNN-ROI may perform better.

In fact, Fig. 3 confirms that when the image semantics is strongly coupled with a specific color pattern, such as on *firework*, *sunset*, and *waterfall*, the ROI locating performance of Diverse Density is apparently better than that of CkNN-ROI. While when the image semantics is not strongly coupled with a specific color pattern, such as on *castle* and *mountain*, the ROI locating performance of CkNN-ROI can be comparable to or even better than that of Diverse Density.

Overall, experiments reported in this section reveal that the efficiency of CkNN-ROI is far better than that of Diverse Density, the retrieval performance and ROI locating performance of CkNN-ROI are often worse than that of Diverse Density, but on many target concepts the performance of CkNN-ROI can be comparable or better. Moreover, considering that Diverse Density can hardly be really used in CBIR systems due to its overwhelmingly high time cost, CkNN-ROI seems like a better choice.

5 Conclusion

CBIR has been widely investigated in the past years. Although many CBIR systems have been developed, few of them can return relevant images where the ROI has been located. On the other hand, although multi-instance learning techniques have been introduced into CBIR, kNN style multi-instance algorithms have not been utilized. In this paper, these two issues are addressed by the CkNN-ROI algorithm, which is a new variant of Citation-kNN and could locate ROI in CBIR with high efficiency.

The experiments reported in this paper are performed on a relatively small image database and a small number of target concepts. Experiments on more images and more target concepts will be performed and reported in the future. Moreover, in the current version of CkNN-ROI, only one ROI can be located in each image and marked with a constant framework. It will be more helpful if all the ROI can be located and marked with a framework that can be with arbitrary shape. This is an interesting issue to be explored in the future.

It is noteworthy that CkNN-ROI is far from a flawless solution to the problem of ROI locating. Nevertheless, it is anticipated that this work might help raise the interest in applying kNN style multi-instance learning algorithms to tasks involving image analysis, and more importantly, arouse investigation on the problem of ROI locating in CBIR.

Acknowledgement

This work was supported by the National Science Foundation of China under the Grant No. 60473046, the Jiangsu Science Foundation under Grant No. BK2005412, and the National 973 Fundamental Research Program of China under the Grant No. 2002CB312002.

References

1. Dasarathy, B.V.: Nearest Neighbor Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos, CA (1991)
2. Maron, O., Lozano-Pérez, T.: A framework for multiple-instance learning. In: Jordan, M.I., Kearns, M.J., Solla, S.A. (eds.): Advances in Neural Information Processing Systems, Vol. 10. MIT Press, Cambridge, MA (1998) 570–576
3. Maron, O., Ratan, A.L.: Multiple-instance learning for natural scene classification. In: Proceedings of the 15th International Conference on Machine Learning, Madison, WI (1998) 341–349
4. Rui, Y., Huang, T.S., Ortega, M., Mehrotra., S.: Relevance feedback: a power tool for interactive content-based image retrieval. IEEE Transactions on Circuits and Systems for Video Technology **8** (1998) 644–655
5. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 1349–1380
6. Wang, J., Zucker, J.-D.: Solving the multiple-instance problem: a lazy learning approach. In: Proceedings of the 17th International Conference on Machine Learning, San Francisco, CA (2000) 1119–1125
7. Yang, C., Lozano-Pérez, T.: Image database retrieval with multiple-instance learning techniques. In: Proceedings of the 16th International Conference on Data Engineering, San Diego, CA (2000) 233–243
8. Zhang, Q., Goldman, S.A.: EM-DD: an improved multi-instance learning technique. In: Dietterich, T.G., Becker, S., Ghahramani, Z. (eds.): Advances in Neural Information Processing Systems, Vol. 14. MIT Press, Cambridge, MA (2002) 1073–1080
9. Zhang, Q., Yu, W., Goldman, S.A., Fritts, J.E. Content-based image retrieval using multiple-instance learning. In: Proceedings of the 19th International Conference on Machine Learning, Sydney, Australia (2002) 682–680
10. Zhou, Z.-H.: Multi-instance learning: a survey. Technical Report, AI Lab, Computer Science & Technology Department, Nanjing University, Nanjing, China, Mar. 2004
11. Zhou, Z.-H., Chen, K.-J., Jiang, Y.: Exploiting unlabeled data in content-based image retrieval. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.): Lecture Notes in Artificial Intelligence, Vol.3201. Springer, Berlin (2004) 525–536
12. Zhou, Z.-H., Jiang, K., Li, M.: Multi-instance learning based web mining. Applied Intelligence **22** (2005) 135–147
13. Zhou, Z.-H., Zhang, M.-L., Chen, K.-J.: A novel bag generator for image database retrieval with multi-instance learning techniques. In: Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence, Sacramento, CA (2003) 565–569