An Efficient Content based Image Retrieval Scheme

Zukuan Wei*[1], Hongyeon Kim[2], Youngkyun Kim[3], Jaehong Kim[4]
[1,2,3] BigData Software Lab., Electronics and Telecommunications Research Institute (ETRI),
Daejeon, 305-700, KOREA,
[4]Dept. of Computer & Information Engineering, YoungDong University, Chungbuk, 370-701, KOREA,
*Corresponding author, e-mail: anlexwee@etri.re.kr[1], kimhy@etri.re.kr[2], kimyoung@etri.re.kr[3],
jhong@youngdong.ac.kr[4]

Abstract

Due to the recent improvements in digital photography and storage capacity, storing large amounts of images has been made possible. Consequently efficient means to retrieve images matching a user's query are needed. In this paper, we propose a framework based on a bipartite graph model (BGM) for semantic image retrieval. BGM is a scalable data structure that aids semantic indexing in an efficient manner, and it can also be incrementally updated. Firstly, all the images are segmented into several regions with image segmentation algorithm, pre-trained SVMs are used to annotate each region, and final label is obtained by merging all the region labels. Then we use the set of images and the set of region labels to build a bipartite graph. When a query is given, a query node, initially containing a fixed number of labels, is created to attach to the bipartite graph. The node then distributes the labels based on the edge weight between the node and its neighbors. Image nodes receiving the most labels represent the most relevant images. Experimental results demonstrate that our proposed technique is promising.

Keywords: Image Retrieval; Image Segmentation; Image Annotation

1. Introduction

In the last decade, due to the improvements in digital photography and storage capacity, there has been rapid growth in the use of digital media such images, video and audio. As the use of digital media increases, effective retrieval and management techniques become more important. Such techniques are required to facilitate the effective searching and browsing of large multimedia databases. In order to respond to this need, image retrieval has been a hot research topic and major technology of many major research projects nearly. Image retrieval systems mainly can be cast in two categories: text based and content based systems. Text-based retrieval is a method that manually annotates images by text. However, this method has many limitations because it is highly labor-intensive, time consuming and unpractical with large databases. In content-based image retrieval (CBIR), the main idea is to extract low level features, such as color, texture and shape, which are used to measure similarity. However, it is difficult to describe high-level semantics using low-level features, thus such image retrieval systems have poor performance for semantic queries. In order to improve the retrieval accuracy of content-based image retrieval systems, research focus has been shifted from designing sophisticated low-level feature extraction algorithms to reducing the "semantic gap" between the visual features and the richness of human semantics.

Automatic image semantic annotation has been approved to be an effective way for realizing semantic image retrievals. In [1], Wang et al. constructed a web image annotation system called 'Annosearch'. The system first searched for semantically and visually similar images on the Web, and then annotations were mined from retrieval results. In [2], Mei et al. proposed a system for annotating by learning semantic distance from each semantic cluster in image set. In [3], Li et al. proposed a new approach of multi-label image annotation for image retrieval based on annotated keywords, a novel annotation refinement approach based on PageRank is also proposed to further improve retrieval performance.

Annotation can be treated as a problem of classification from a point of view of machine learning. Some previous annotation work are basing on k-nearest neighbors (k-NN) [4, 5]. In the field of classification, k-NN is a relative low precision classifier because it lacks training process.

Received June 7, 2013; Revised August 3, 2013; Accepted August 15, 2013

Thus, classification approach with higher precisions should be used in annotation process. Moreover, for the abundant and various visual content of image, multi-labels should be annotated to get more precise description of image in semantic level. What's more, most of the proposed approaches are assumed that the database is constant. But for most practical databases (like internet image collections), new images are constantly added, in this case, these approaches won't perform well, primarily due to its resource intensive matrix computations.

Most image retrieval systems are proposed assuming that the databases are constant. But for most practical databases (like internet image collections), new images are constantly added to the image collection that poses a considerable challenge, primarily due to its resource intensive matrix computations. An incremental variant of pLSA proposed by Wu et al. [6] tried to improve of the computational efficiency. However they failed to address the issue of storage complexity. In [7], Chandrika Pulla et al. proposed a novel method based on a bipartite graph model can address these issues well, but has a low precision.

In this paper, we propose a new model for content-based image retrieval. Firstly, the images are automatically annotated with several labels, and then the labels are used to build a bipartite graph, after that we can use it to retrieve relevant images at runtime. The rest of this paper is structured as follows: Section 2 discusses the image annotation process. Section 3 presents the bipartite graph model we build and Section 4 presents the retrieval process. Experiments and analysis will be presented in Section 5 and conclusions will be drawn in Section 6.

2. Image Annotation

2.1. Image Annotation

As it has already been mentioned, automatic image semantic annotation is an effective way for realizing semantic image retrieval. In this section we will discuss the role of image segmentation and annotation, and we will present the approach used in this work.

Given the entire set of images of a given database and their extracted low-level features, it may easily be observed that regions that correspond to the same concept have similar low-level descriptions. Also, images that contain the same high-level concepts are typically consisted of similar regions. For example, regions that contain the concept 'sky' are generally visually similar, i.e. the color of most of them should be some tone of 'blue'. On the other hand, images that contain 'sky' often ate consisted of similar regions.

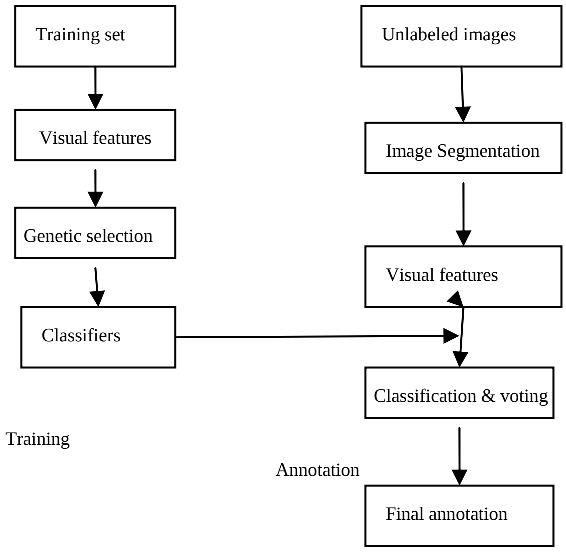

Figure 1. Framework of Image Annotation

The aforementioned observations indicate that similar regions often co-exist with some high-level concepts. This means that region co-existences should be able to provide visual descriptions which can discriminate between the existences or not of certain high-level concepts. By appropriately quantizing the regions of an image dataset, we can extract efficient descriptions. Thus, we can create a set of labels of the most common region types and use multiple labels to represent an image. In this paper, we use an approach proposed in [3] to annotate the unlabeled images. The framework of annotation approach is shown in Figure 1. As it can be seen, the process contains two main stages: training and annotation.

At training stage, Support Vector Machines (SVM) is used as basic classifiers, for the reason that SVM classifiers can be reused in incremental database while measures like K-Means cluster can't. In this paper, the classification problem including multi-classes can be seen as a set of two-class classification. Thus, a SVM is trained for every couple of classes in the set. Assuming there are k classes in the training set, ***k*(k-1)/2*** SVMs need to be trained totally. Then visual features like color, color layout, color structure, homogeneous texture, edge histogram and region shape are extracted to represent images. However, if all these features are involved in the classification, the dimension complexity will bring high computational costing. Meanwhile, different features always have different importance, thus the weights of them need to be automatically adjusted in classification. In order to improve the precision while decrease the complexity of features, the mechanism of genetic selection is introduced to select best features for classification between two classes during the training process.

Two kinds of chromosomes are used here: a real chromosome { w_i }, $w_i \in [0,1], i=1,2,\ldots,n$ represents the weights corresponding to feature descriptors; a binary chromosome { a_i }, $a_i \in \{0,1\}, i=1,2,\ldots,n$ represents the presence or absence of a feature descriptor in the optimal feature subset, n is the number of features. For every one vs. one SVM, the fitness function in GA is designed as the classification accuracy in training set. More details about bi-coded GA can be seen in [6]. Genetic operation is terminated when the process reaches the maximum number of generation; the fittest individual is treated as the optimal selection result. The relative optimal weighted feature set is V: $V=\{ v_i \}, v_i = w_i \times a_i \times f_i, i=1,2,\ldots,n,$ f_i is the i-th descriptor. After training, we have optimal subsets and corresponding weights for every couple of classes in training set.

At annotation stages, all the images in the database are segmented based on their low-level coherence of features, such as gray-level similarity and texture etc. For each image region, each one vs. one SVM classifier is used to decide a label according the optimal feature subset and optimal weights trained before. The final class label is decided using a majority voting approach, every SVM vote the unlabeled region with its classification result, its final classification result $f(x)$ is the label of g_i which has highest voting score:

$$f(x) = \arg\max g_i(x), \quad i = 1,2,\ldots,k \tag{1}$$

Each region is annotated with semantic label voted maximum based on the annotation algorithm, and then all the region labels are merged as the final image annotation.

2.2. Refinement

The performance of annotation may be influenced by many factors, one of which is the confine of segmentation algorithm. Because some regions don't have specific meanings, and annotate these regions will bring annotation errors. Consequently, the image retrieval results will be influenced. So an annotation refinement approach is needed to rank the candidate annotations and get rid of irrelevant annotations. Although mangy algorithms [8-12] satisfy the requirement [8], the algorithm using Random Walk with Restarts (RWR) is chosen to re-rank the candidate annotations for its simplicity and effectiveness in this paper.

To facilitate the annotation refinement process, a confidence score for the candidate annotations should be provided. The probability that a region is involved in a label class is often used as the confidence score. More specifically, the confidence score of label w_i is defined as:

$$score\,(w_i) = p\,(w_i \mid I) \tag{2}$$

In order to fully utilize the confidence scores of the candidate annotations and the corpus information, we reformulate the image annotation refinement process as a graph ranking problem and solve it with the RWR algorithm. Each candidate annotation $\boldsymbol{v_i}$ is considered as a vertex of a graph G. All vertices of G are fully connected with proper weights. The weight of an edge is defined based on the ‘co-occurrence” similarity as follows:

$$.sim(w_i, w_j) = \begin{cases} \dfrac{num(w_i, w_j)}{\min(num(w_i), num(w_j))} & num(w_i, w_j) > 0 \\ 0 & num(w_i, w_j) = 0 \end{cases} \tag{3}$$

The RWR algorithm performs as follows [13]. Assume that there is a random walker that starts from node $\boldsymbol{w_i}$ with certain probability. At each time-tick, the walker has two choices. One is to randomly choose an available edge to follow. The other choice is to jump to $\boldsymbol{w_i}$ with probability ***cxv(j)***, where ***v*** is the restart vector and ***c*** is the probability of restarting the random walk [13].

Assume that ***G*** is a graph with ***N*** vertices $\boldsymbol{w_i}$ constructed as aforementioned description. Let ***A*** be the adjacency matrix of ***G***. ***A*** is column-normalized to ensure that the sum of each column in ***A*** is one. The original confidence scores of candidate annotations are considered as the restart vector ***v***. ***v*** is normalized such that the sum of all elements in ***v*** is one. The aim is to estimate the steady state probability of all vertices, which is denoted by ***u***. Let ***c*** be the probability of restarting the random walk. It is empirically set to be 0.3 in our implementation. Then the N-by-1 steady state probability vector ***u*** satisfies the following equation:

$$\boldsymbol{u = (1 - c) Au + cv} \tag{4}$$

Therefore,

$$\boldsymbol{u = c (I - (1 - c) A)^{-1} v} \tag{5}$$

where ***I*** is the ***NxN*** identity matrix.

The $\boldsymbol{i_{th}}$ element ***u(i)*** of the steady state vector ***u*** is the probability that $\boldsymbol{w_i}$ can be the final annotation.

After the rank score of every word is obtained, there are two methods to decide the final annotations. One way is to rank the score from high to low, and then choose a particular number of them. The other way is to choose all the annotations with probabilities larger than a threshold and the others are omitted.

3. Bipartite Graph Model

When the stage of image annotation is finished, we should maintain the relationship between images and labels. In real applications, the region types are usually too many, and each image has only of a few of them, so the matrix of the images and labels is sparse, and it is a waste of storage, we can use a bipartite graph model to convert the matrix into a bipartite graph of images and labels, and it can address the storage problem efficiently. The structure of a BGM is showed in Figure 2.

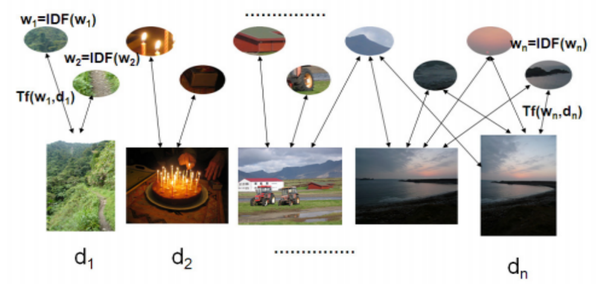

Figure 2. The Structure of a BGM

The edges between images and labels are weighted with frequencies of labels in the images and each is also associated with an inverse image frequency value. These values determine the importance of a label to a particular image. G = (W, D, E) is the bipartite graph such that W = {$\boldsymbol{w_1}$, $\boldsymbol{w_2}$ ... , $\boldsymbol{w_n}$ }, I = {$\boldsymbol{i_1}$, $\boldsymbol{i_2}$… , $\boldsymbol{i_n}$} and ***E*** **={** $e_{w_1}^{i_1}$…, $e_{w_n}^{i_1}$ …, $e_{w_n}^{i_m}$ }. Here the weight associated with $\boldsymbol{w_1}$ **= *IDF*** ($\boldsymbol{w_1}$) and that of $e_{w_1}^{i_1}$ = ***TF (***$\boldsymbol{w_1}$***,*** $\boldsymbol{i_1}$***)***.

An image may contain many labels and a label may be present in many images. Similarity of two images can be measured in based on the number of labels they share. If images A and B as well as A and C are similar, then B and C are also similar. This gets reflected in the paths which traverse the graph from image to labels and then back to images.

4. Image Retrieval

When a query image is given, an image segmentation algorithm is run to segment it into several regions. Each region is classified by our pre-trained SVMs, and a corresponding label is distributed to it. The final labels of the query image are obtained by merging all the region labels. Then a query node, a kind of image node, is created attaching itself to the label nodes that represent the labels it has. The node then distributes the labels based on the edge weight between the node and its neighbors, such that the received amount of labels is directly proportional to the edge weight. The query node is disconnected from the graph. The neighbors then propagate the labels to their neighbors. If the node is a image node, the distribution of the labels among its edges is determined according to the quantity which is proportional to the flow capacity calculated by the normalized Term Frequency (TF) value. If the node is a word node, then a penalty, which is proportional to the Inverse document Frequency (IDF) value of the word, is taken from the amount of label it receives and the rest is distributed like the document node based on the flow capacity of its edges. Hence higher the edge weights the more label is propagated to the relevant node. At each node the label is compared with a cutoff value which is the least amount of the label needed for a node to forward the label. Hence the label is propagated to relevant documents and terms until a cutoff value is reached at which point label is no longer propagated. The nodes receiving the most labels are the most relevant images. Thus, it divides the nodes in the bipartite graph into relevant and non-relevant sets similar to a graph cut algorithm.

Our system also supports keywords queries. Under this circumstance, the users are asked to input keywords as queries and the issue of image retrieval is transferred into the issue of text-based retrieval. It is faster than image queries, because the process of segmenting images and obtaining labels is not needed. We only need to propagate the labels between label nodes and image nodes, without creating a new query node.

5. Experimental Results on Image Retrieval

The most common evaluation measures used in IR are precision and recall, usually presented as a precision vs. recall graph. Researchers are familiar with PR graphs and can extract information from them without interpretation problems. Precision and recall are defined as a bellow: x`

$$\mathbf{precision} = \frac{\textit{No.relevant images retrived}}{\textit{Total No.images retrived}}$$

$$\mathbf{recall} = \frac{\textit{No.relevant images retrieved}}{\textit{Total No.relevant images in the database}}$$

Precision and recall are standard measures in IR, which give a good indication of system performance. Either value alone contains insufficient information. We can always make recall, simply by retrieving all images. Similarly, precision can be kept high by retrieving only a few images. Thus precision and recall should either be used together, or the number of images retrieved should be specified.

In our experiment, we select more than 1400 images from Pascal dataset as experimental dataset. It contains 15 object classes. Due to objects contained, every image has 1 to 5 labels. At the training stage, for objects in every category, classifiers are constructed. At annotation stage, whole image is firstly segmented using image segmentation algorithm, and then annotated by pre-trained SVMs. Normalized cut algorithm is used for image segmentation. Considering objects contained in images, we set N=8.

Table 1 shows the average precision and recall of 15categories based on image annotation and image annotation refinement approach proposed in this paper. ML means image annotation without annotation refinement, while ML-refinement means image annotation with annotation refinement.

Table 1. Result in Precision and Recall

	recall	precision
ML annotation	0.463	0.122
ML-refinement	0.225	0.134

6. Conclusion

In this paper, we propose a new scheme for content-based image retrieval. In the annotation stage, a bi-coded genetic algorithm is applied to select optimal feature subset and corresponding optimal weights for every one vs one SVM classifier. For every region of images, optimal weighted feature subset and annotation refinement improve the precision of annotation. Then the reserved labels are used to build a bipartite graph and it will be used in the retrieval process.

However, image annotation based on image segmentation algorithms is often unsatisfactory, for the reason that they often produce regions that don’t have specific meanings, thus annotate these regions will bring annotation error. In future, continuous efforts in inventing new annotation algorithms to obtain dedicated annotations will be necessary.

Acknowledgements

This work was supported by the IT R&D program of MKE/KEIT, KOREA. [10038768, the Development of Supercomputing System for the Genome Analysis]

References

[1] X Wang, L Zhang, et al. AnnoSeatch: *Image auto-annotation by search*. Proceeding of IEEE Computer Vision and Pattern Recognition. 2006; 1483-1490.

[2] T Mei, Y Wang, XS Hua, S Gong, S Li. *Coherent image annotation by learning semantic distance.* proceedings of Conference on Computer Vision and Pattern Recognize. 2008; 1-8.

[3] R Li, YF Zhang, Z Lu, J Lu, Y Tian. *Technique of Image Retrieval based on Multi-label Image Annotation.* Second International Conference on Multimedia and Information Technology. 2010; 10-13.

[4] X Li, L Chen, L Zhang, F Lin, WY Ma. *Image annotation by large-scale content-based image retrieval.* Proceeding of the 14th Annual ACM international Conference on Multimedia. 2006; 607-610.

[5] TM Hamdani, AM Alimi, F Karray. *Distributed Genetic Algorithm with Bi-Coded Chromosomes and a New Evaluation Function for Features Selection.* Proceeding of the IEEE Congress on Evolutionary Computation, Canada. 2006: 581-588.

[6] H Wu, Y Wang, X Cheng. *Incremental probabilistic latent semantic analysis for automatic question recommendation*. Proc. ACM Conf. Recommender Systems, ACM Press. 2008; 99-106.

[7] Chandrika Pulla, Suman Karthik, CV Jawahar. *Efficient Semantic Indexing for Image Retrieval.* International Conference on Pattern Recognition. 2010: 3276-3279.

[8] C Wang, F Jing, L Zhang, HJ Zhang. *Image Annotation Refinement using Random Walk with Research.* Proceedings of the 14th Annual ACM international Conference on Multimedia. Santa Barbara, California, USA. 2006.

[9] E Chang, G Kingshy, G Sychay, G Wu. CBSA: *content-based soft annotation for multimodal image retrieval using Bayes point machines*. IEEE Trans. On CSVT. 2003; 13(1): 26-38.

[10] Chandrika Pulla, Suman Karthik, CV Jawahar. *Efficient Semantic Indexing for Image Retrieval.* International Conference on Pattern Recognition. 2010: 3276-3279.

[11] Yohan Jin, Ltifur Khan, B Prabhakaran. *To be Annotated or not? the Randomized Approximation Graph Algorithm for Inage Annotation Refinement Problem.* ICDE2008. 2008.

[12] Wong RCF, Leung CHC. Automatic Semantic Annotation of Real-World Web Images. *IEEE Transaction on Pattern Analysis and Machine Intelligenc*e. 2008; 30(11): 1933-1944.

[13] Page L, Brin S, Winograd T. *The PageRank Citation Ranking: Bringing Order to the web*. Technical Report, Stanford University, Stanford, CA. 1988.