b152591

A COMBINATION OF GRAPH RANKING AND TERM COLLOCATION INFORMATION FOR IMAGE ANNOTATION

Krittapad Suriya

A Thesis Submitted in Partial
Fulfillment of the Requirements for the Degree of
Master of Science (Computer Science)
School of Applied Statistics
National Institute of Development Administration
2006

A COMBINATION OF GRAPH RANKING AND TERM COLLOCATION INFORMATION FOR IMAGE ANNOTATION

Krittapad Suriya

School of Applied Statistics

The Examining Committee Approved This Thesis Submitted in Partial Fulfillment of the Requirements for the Master Degree of Science (Computer Science)

Assistant Professor .... Committee Chairman
(Ohm Sornil, Ph.D.)

Associate Professor .... Committee
(Pipat Hiranvanichakorn, D.E.)

Associate Professor .... Committee
(Surapong Auwatanamongkol, Ph.D.)

Associate Professor .... Acting Dean
(Surapong Auwatanamongkol, Ph.D.)

Date ....

ABSTRACT

Title of Thesis	:	A Combination of Graph Ranking and Term Collocation Information for Image Annotation
Author	:	Mr. Krittapad Suriya
Degree	:	Master of Science (Computer Science)
Year	:	2006

In a large image database, an automatic image annotation plays an important role to assign a caption to an image. There are many applications of image annotation which include information extraction, knowledge acquisition, finding answers from specific questions, etc.

In this research, we propose an image annotation approach using a graph ranking algorithm on a novel structure which includes similarities among regions within images and caption term collocation information. Caption terms are selected using a graph random walk method biased toward the query image. A series of experiments are performed on standard image data sets to evaluate the proposed approach. The results show the annotation accuracy of 56.96%.

ACKNOWLEDGEMENTS

To make this thesis possible I wish to thanks many people who give helpful, valuable contribution and suggestion that make this thesis possible. First of all, I would like to express my deepest appreciation to my major advisor, Assistant Professor Dr. Ohm Sornil, for valuable guidance, and for encouragement and support throughout the conduct and writing in completing this thesis. Moreover, I am deeply grateful to the committee members, Associate Professor Dr. Pipat Hiranvanichakorn and Associate Professor Dr. Surapong Auwatanamongkol, for their valuable advice and constructive comments.

Second, I am also grateful to all of the lecturers, classmates and officers of the School of Applied Statistics, and the Department of Computer Science, who spared their precious time to give suggestions and assistance concerning this thesis.

Finally, I wish to special my gratitude to my parents and brother for their support, love and understanding over the time.

Krittapad Suriya
April 2007

TABLE OF CONTENTS

LIST OF TABLES

Tables **Page**

LIST OF FIGURES

Figures **Page**

CHAPTER 1

INTRODUCTION

Since collections of images grow at a very fast rate, from stock photo collections and proprietary databases to collections on the World Wide Web, the size of image database has been continuously increasing. Therefore, the ability to search and organize the image is essential to the huge image database. There are several researches which proposed techniques for retrieving images based on their contents such as search by image features, search by segment features, search for specific types of images and search by image sketch. These approaches require the user to know about image concepts like color or texture which is not familiar to most people. One solution to this problem is to annotate such images manually and then search those annotations. However, the solution is expensive and tedious. Hence, there has been great effort to annotate the image automatically.

The automatic image annotation or caption is an approach to predicting words for image. In auto-annotation, it attempts to find associations between image features and text. Several automatic annotation methods applied machine learning techniques to find the annotated training images. Then, the discovered associations are applied to annotate new images.

In this thesis, we introduce a novel graph structure which includes similarities among regions within images and caption term collocation information; from the graph structure, caption terms are selected using a graph random walk method biased toward the query image.

A modified Random Walk with Restarts algorithm is also proposed to improve performance of automatic image annotation.

In Chapter 2, some related works in image annotation are discussed. The proposed technique for automatic image annotation is presented in Chapter 3. Chapter 4 describes the experimental results of performance evaluation. In Chapter 5, the work carried out for this thesis is summarized and conclusions are given.

CHAPTER 2

RELATED WORK

A number of methods have been proposed for image annotation. Maron and Ratan (1998: 341-349) use multiple instances learning to train classifiers to identify particular keywords from image data using labeled bags of examples. In their approach, an image is a "positive" example if it contains a particular object (e.g. tiger) in the image, but "negative" if it doesn't. Wenyin, Dumais, Sun, Zhang, Czerwinski and Field (2001: 326-333) proposed a semi-automatic strategy for annotating images using the user's feedback of a retrieval system. The query keywords which receive positive feedback are collected as possible annotation to the retrieved images. Li and Wang (2003: 1075-1088) modeled image concepts by 2-D multiresolution Hidden Markov Models and labeled an image with the concepts best fit the content.

Carson, Belongie, Greenspan and Malik (2002: 1026-1038) retrieved images from large and varied collections using image content as a key. A new image representation as also proposed. It provides a transformation from the raw pixel data to a small set of image regions those are coherent in color and texture. Blobworld representation is created by clustering pixels in a joint color-texture-position feature space. This paper describes a system that uses the Blobworld representation to retrieve images from this collection. An important aspect of the system is that the user is allowed to view the internal representation of the submitted image and the query results.

Chen, Gan and Suel (2002: 549-557) studied I/O-efficient techniques to perform this iterative computation. They derived two algorithms for Pagerank based on techniques proposed for out-of-core graph algorithms, and compared them to two existing algorithms proposed by Haveliwala (2002: 517-526). They also considered the implementation of a recently proposed topic-sensitive version of Pagerank. Their experimental results showed that for very large data sets, significant improvements over previous results can be achieved on machines with moderate amounts of memory. On the other hand, at most minor improvements are possible on data sets that are only moderately larger than memory, which is the case in many practical scenarios.

Pan, Yang, Faloutsos and Duygulu (2004) proposed a Graph-based Automatic Image Caption (GCap) method for auto caption. It is in fact more general, capable of attacking the general problem of finding correlations between arbitrary modalities of arbitrary multimedia collections. In auto-captioning, it finds correlations between two modalities, image features and text. In a more general setting, such as video clips, GCap can be easily extended to find correlations between some other modalities, like, the audio parts and the image parts.

Other work models directly the association between words and the numerical features of the regions, for example, the generative hierarchical aspect model (Barnard and Forsyth, 2001: 408-415; Barnard, Duygulu, Freitas, Forsyth, Blei and Jordan, 2003: 1107-1135), and the contextual model which models spatial consistency by Markov random field (Carbonetto, Freitas and Barnard, 2004: 350-362). These methods try to find the actual association between image regions and terms for image annotation and for a greater goal of object recognition.

CHAPTER 3

THE PROPOSED TECHNIQUE

In this section, the proposed technique for image annotation is described. The information about how image regions are associated with terms is established from a captioned image set. Each image in a captioned image set is annotated with terms describing the image content. Captioned images can come from many sources.

3.1. Image Annotation Process

The Image Annotation process of our model is shown in figure 3.1

The model is graph-based model. The graph-based model is supervised annotation using graph-based features to train on Query Sensitive Random Walk (QSWR). The goal of the process is to select a few segments as an annotation that can represent the original image although the result of each annotation model is different caption followed by each functional model.

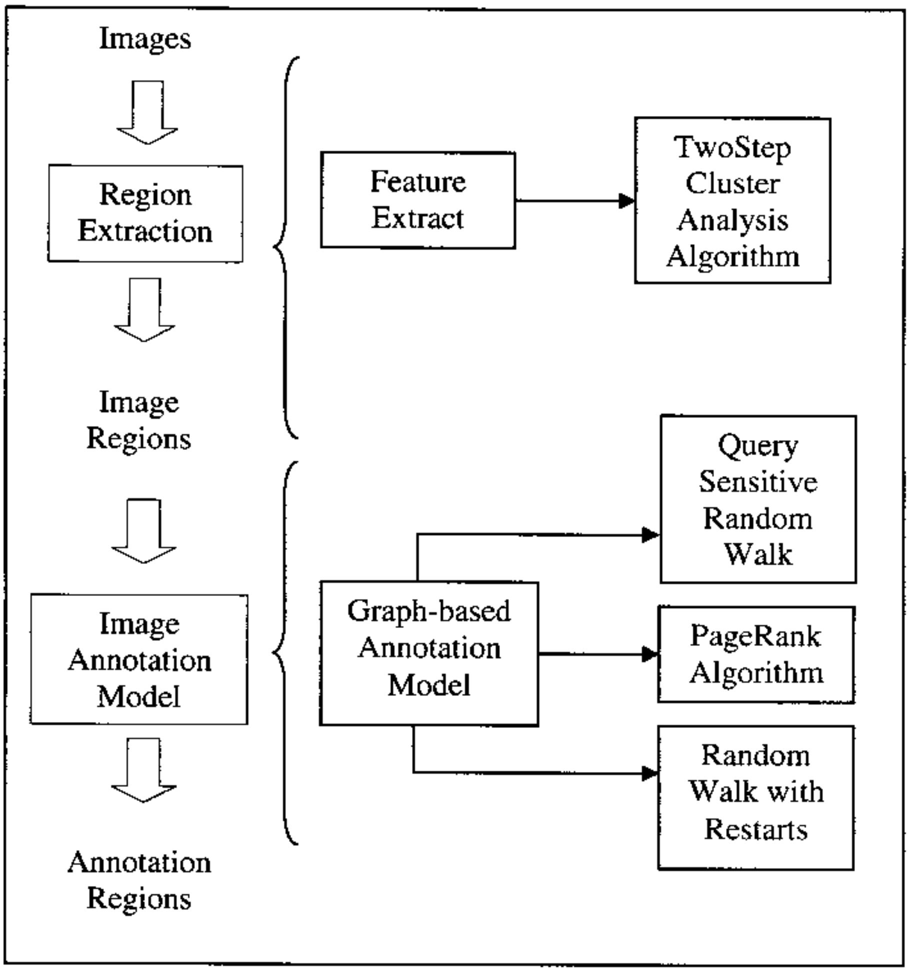


Figure 3.1 Overview of Image Annotation Process

3.2. Data Preprocessing

Creating image regions includes three steps

1. Extract color, texture, and position features for each pixel at a selected scale.

2. Group pixels into regions by modeling the distribution of pixel features using TwoStep clustering algorithm.

3. Create a representative vector for each region.

3.2.1 Extracting Color Features

Each image pixel has a three-dimensional color descriptor in the L*a*b* color space (Chang and Lee, 2007). This color space is approximately perceptually uniform; thus, distances in this space are meaningful. We smooth the color features as discussed in Section 3.2.3 in order to avoid over segmenting regions such as tiger stripes based on local color variation; otherwise, each stripe would become its own region.

Color is perceived by human as combinations of tristimuli R (Red), G (Green) and B (Blue) or three primary colors. It is important to choose a good color space for color image quantization, since the distance measure in color space must conform to human visual perception.

CIE color space (Wyszecki and Styles, 1982: 130-173) was developed to represent perceptual uniformity. Furthermore, in CIE color space, color difference can be calculated as the Euclidean distance between two color points. CIE color space has three primaries, X, Y and Z. Colors can be represented by combinations of X, Y and Z. The X, Y and Z values can be computed from RGB tristimulus coordinates using a linear transformation as show:

$$X = (0.412453*R + 0.357580*G + 0.130423*B)/(242.36628)$$

$$Y = (0.212671*R + 0.715160*G + 0.072169*B)/(255)$$

$$Z = (0.019334*R + 0.119193*G + 0.950227*B)/(277.63277)$$

By change XYZ to LAB

$$L^* = 116 \; f\left(Y/Y_n\right) - 16$$

$$a^* = 500 \; \left[f\left(X/X_n\right) - f\left(Y/Y_n\right)\right]$$

$$b^* = 200 \; \left[f\left(Y/Y_n\right) - f\left(Z/Z_n\right)\right]$$

where

$$f(t) = \begin{cases} t^{1/3} & \text{for} \quad t < 0.008856 \\ 7.787 \quad t + 16 \; /116 & \text{other} \end{cases}$$

Here X_n, Y_n and Z_n are the CIE XYZ tristimulus values of the reference white point.

The three parameters, L, a, and b in CIE color space represent the brightness of the color, its position between magenta and green, and its position between yellow and blue, respectively.

3.2.2 Extracting Texture Features

Texture is a well-researched property of image regions and many texture descriptors have been proposed, including multiorientation filter banks and the second-moment matrix. We will not elaborate here on the classical approaches to texture segmentation and classification, both of which are challenging and well-studied tasks. Rather, we introduce a new perspective related to texture descriptors and texture grouping motivated by the content-based retrieval task.

Whereas color is a point property, texture is a local neighborhood property. It does not make sense to talk about the texture of zebra stripes at a particular pixel without specifying a neighborhood around that pixel. In order for a texture descriptor to be useful, it must provide an adequate description of the underlying texture parameters and it must be computed in a neighborhood which is appropriate to the local structure being described.

The image represents the coordinates in the texture space we have 3 methods, as follows:

- Anisotropy (GaËrding and Lindeberg, 1996: 163-191) is a method of enhancing the image quality of textures on surfaces that are far away and steeply angled with respect to the camera.

$$a = 1 - \lambda_2 / \lambda_1 \tag{3.1}$$

- Polarity (Freeman and Adelson, 1991: 891-906, Granlund and Knutsson, 1995: 399-412) is a measure of the extent to which the gradient vectors in a certain neighborhood all point in the same direction.

$$p_\delta = \frac{|E_+ - E_-|}{E_+ + E_-} \tag{3.2}$$

The definitions of E_+ and E_- are measures of how many gradient vectors in the window.

$$E_+ = \sum_{x,y} G_\delta(x,y)\left[\nabla I \cdot \hat{n}\right]_+ \tag{3.3}$$

and

$$E_- = \sum_{x,y} G_\delta(x,y)\left[\nabla I \cdot \hat{n}\right]_- \tag{3.4}$$

- Contrast (GaËrding et al., 1996: 163-191) is a method of enhancing the image quality of textures.

$$c = 2\sqrt{\lambda_1 + \lambda_2}^{\,3} \tag{3.5}$$

3.2.3 Combining Color, Texture, and Position Features

The final color/texture/position descriptor for a given pixel consists of eight values: three for color, three for texture and two for position. The three color components are the L*a*b* coordinates found after spatial averaging using a mixture of Gaussians at the selected scale. The three texture components are ac, pc, and c, computed at the selected scale; the anisotropy and polarity are each modulated by the contrast since they are meaningless in regions of low contrast. The two position components are x and y pixel position at the select location. For example, the eight values feature as a pixel feature vector shown in Table 3.1. In effect, a given textured patch in an image first has its texture properties extracted and then is replaced by a smooth patch of averaged color. In this manner, the color and texture properties in a given region are decoupled; for example, a zebra becomes a gray horse plus stripes.

In order to segment each image automatically, we model the joint distribution of color, texture, and position features with a TwoStep cluster. We use the TwoStep cluster algorithm to separate of this model; the resulting pixel-cluster memberships provide a segmentation of the image. After the image is segmented into regions, a description of each region's color, texture and position characteristics is produced. In a querying task, the user can access the regions directly, in order to see the segmentation of the query image and specify which aspects of the image are important to the query.

Table 3.1 Pixel Features

	L	a	b	Anisotropy	Polarity	Contrast	Position of X	Position of Y
Pixel 1								
Pixel 2								
Pixel 3								
Pixel 4								
Pixel 5								
Pixel m-2								
Pixel m-1								
Pixel m								

From segments of each image, we can prepare data which represent properties for each segment. The process begins with images from the segment are passed through a series of preprocessing steps shown in Figure 3.2.

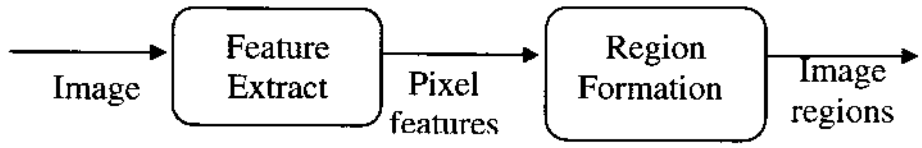


Figure 3.2 Preprocessing Operations

Each segment composes of groups of regions separated by spacing delimitations. The groups of regions are then transcribed to get segment terms. These segment terms are adapted in the annotation model as fundamental foundation that the semantics of the images can be naturally expressed through sets of segment regions.

3.3 TwoStep Cluster Analysis Algorithms

The TwoStep cluster method (Clementine®, 2003: 53-59) is a scalable cluster analysis algorithm designed to handle very large data sets. It can handle both continuous and categorical variables and attributes. It requires only one data pass which contains two steps as follows.

3.3.1 Pre-cluster

The pre-cluster step uses a sequential clustering approach. It scans the data records one at a time and decides if the current record should be merged with the previously formed clusters or starts a new cluster based on the distance criterion (described below).

The procedure is implemented by constructing a modified cluster feature (CF) tree show in Figure 3.3. The CF tree consists of levels of nodes, and each node contains a number of entries. A leaf entry (an entry in the leaf node) represents a final sub-cluster. The non-leaf nodes and their entries are used to find a correct leaf node that a new record is to be added. Each entry is characterized by its CF that consists of the entry's number of records, mean and variance of each range field, and counts for each category of each symbolic field. For each successive record, starting form the root node, it is recursively guided by the closest entry in the node to find the closest child node, and descends along the CF tree. Upon reaching a leaf node, it finds the closest leaf entry in the leaf node. If the record is within a threshold distance of the closest leaf entry, it is absorbed into the leaf entry and the CF of that leaf entry is updated. Otherwise it starts its own leaf entry in the leaf node. If there is no space in the leaf node to create a new leaf entry, the leaf node is split into two. The entries in the original leaf node are divided into two groups using the farthest pair as seeds, and the remaining entries are redistributed based on the closeness criterion.

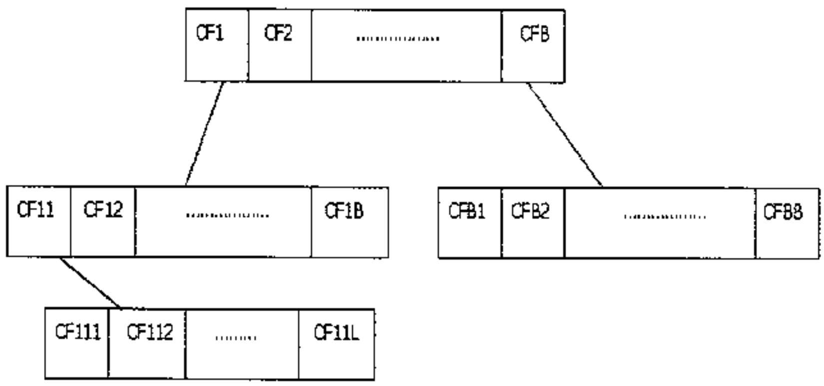


Figure 3.3 Cluster Feature Tree

If the CF tree grows beyond allowed maximum size, the CF tree is rebuilt based on the existing CF tree by increasing the threshold distance criterion. The rebuilt CF tree is smaller and hence has space for new input records. This process continues until a complete data pass is finished. For details of CF tree construction, see the BIRCH algorithm (Zhang, Ramakrishnon and Livny, 1996: 103-114)

All records added to the same entry can be collectively represented by the entry's CF. When a new record is added to an entry, the new CF can be computed using the content of this new record and the old CF without knowing the set of current individual records in the entry. These properties of CF make it possible to maintain only the entry CFs, rather than the sets of individual records. Hence the CF-tree is much smaller than the original data and can be stored in memory more efficiently.

Note that the structure of the constructed CF tree may depend on the input order of the cases or records. To minimize the order effect, data the records are rearranged in random order before building the model.

3.3.2 Clustering

The cluster step takes sub-clusters (non-outlier sub-clusters if outlier handing is used) resulting from the pre-cluster step as input and then groups them into the desired number of clusters. Since the number of sub-clusters is much less than the number of original records, traditional clustering methods can be used effectively. TwpStep uses an agglomerative hierarchical clustering method, because it works well with the auto-cluster method (see the section on auto-clustering below).

Hierarchical clustering refers to a process by which clusters are recursively merged, until at the end of the process only one cluster remains and contains all records. The process starts by defining a starting cluster for each of the sub-clusters produced in the pre-cluster step. (For more information, see "Pre-cluster" on page 11.) All clusters are then compared, and the pair of clusters with the smaller distance between them is selected and merged into a single cluster. After merging, the new set of clusters is compared, the closest pair is merged, and the process repeats until all clusters have been merged. (If you are familiar with the way a decision tree is built, this is a similar process, except in reverse.) Because the clusters are merged recursively in this way, it is easy to compare solutions with different numbers of clusters. To get a five-cluster solution, simply stop merging when there are five clusters left; to get a four-cluster solution, task the five-cluster solution and perform one more merge operation, and so on.

3.3.3 Distance Measure

The TwoStep clustering method uses a log-likelihood distance measure, to accommodate both symbolic and range fields. It is a probability-based distance. The distance between two clusters is related to the decrease in log-likelihood as they are combined into one cluster. In calculating log-likelihood, normal distributions for range fields and multinomial distributions for symbolic fields are assumed. It is also assume that the fields are independent of each other, and so are the records. The distance between clusters i and j is defined as

$$d(i,j) = \xi_i + \xi_j - \xi_{\langle i,j \rangle} \tag{3.7}$$

where

$$\xi_v = -N_v \left(\sum_{k=1}^{K^A} \frac{1}{2} \log \left(\hat{\sigma}_k^2 + \hat{\sigma}_{vk}^2 \right) + \sum_{k=1}^{K^B} \hat{E}_{vk} \right) \tag{3.8}$$

and

$$\hat{E}_{vk} = -\sum_{l=1}^{L_k} \frac{N_{vkl}}{N_v} \log \frac{N_{vkl}}{N_v} \tag{3.9}$$

In these expressions,

K^A is the number of range type input fields,

K^B is the number of symbolic type input fields,

L_k is the number of categories for the k^{th} symbolic field,

N_v is the number of records in cluster v,

N_{vkl} is the number of records in cluster v which belongs to the l^{th} category of the k^{th} symbolic field ,

$\hat{\sigma}_k^2$ is the estimated variance of the k^{th} continuous variable for all records,

$\hat{\sigma}_{vk}^2$ is the estimated variance of the k^{th} continuous variable for records in the v^{th} cluster, and

$\langle i, j \rangle$ is an index representing the cluster formed by combining clusters i and j.

If $\hat{\sigma}_k^2$ is ignored in the expression for ξ_v, the distance between clusters i and j would decrease in log-likelihood when the two clusters are combined. The $\hat{\sigma}_k^2$ term is added to solve the problem caused by $\hat{\sigma}_k^2 = 0$, which would result in the natural logarithm being undefined. (This can occur, for example, when a cluster has only one case.)

3.3.4 Number of Clusters (auto-clustering)

TwoStep can use the hierarchical clustering method in the second step to assess multiple cluster solutions and automatically determine the optimal number of clusters for the input data. A characteristic clustering is that it produces a sequence of partitions in one run: 1, 2, 3, ... clusters. In contrast, a k-means algorithm would need to run multiple times (one for each specified number of clusters) in order to generate the sequence. To determine the number of clusters automatically, TwoStep uses a two-stage procedure that works well with the hierarchical clustering method. In the first stage, The BIC for each number of clusters within a specified range is calculated and used to find the initial estimate for the number of clusters.

The BIC is computed as

$$BIC(J) = -2\sum_{j=1}^{J}\xi_j + m_J \log(N) \tag{3.10}$$

where

$$m_J = J\left\{2K^A + \sum_{k=1}^{K^B}(L_K - 1)\right\} \tag{3.11}$$

And other terms defined as in "Distance Measure" above. The ratio of change in BIC at each successive merging relative to the first merging determines the initial estimate. Let $dBIC(J)$ be the difference in BIC between the model with J clusters and that with $(J+1)$ clusters, $dBIC(J) = BIC(J) - BIC(J+1)$. Then the change ratio for model J is

$$R_1(J) = \frac{dBIC(J)}{dBIC(1)} \tag{3.12}$$

If $dBIC(1)<0$, then the number of clusters is set to i (and the second stage is omitted). Otherwise, the initial estimate for number of clusters k is the smallest number for which for which $R_1(J) < 0.04$.

In the second stage, the initial estimates are refined by finding the largest relative increase in distance between the two closest clusters in each hierarchical clustering stage. The second stage are performed as follows:

Staring with the model C_k indicated by the BIC criterion, take the ratio of minimum inter-cluster distance for that model and the next large model C_{k+1}, that is , the previous model in the hierarchical clustering procedure,

$$R_2(k) = \frac{d_{\min}(C_k)}{d_{\min}(C_{k+1})} \tag{3.13}$$

where

C_k is the cluster model containing k clusters and

$d_{\min}(C)$ is the minimum inter-cluster distance for cluster model C.

Now from model C_{k-1}, compute the same ratio with the following model C_k, as above. Repeat for each subsequent model until the ratio $R_2(2)$ has been computed.

Compare the two largest R_2 ratios; if the largest is more that 1.15 times the second largest, then select the model with the largest R_2 ratio as the optimal number of clusters; otherwise, from those two models with the largest R_2 values, select the one with the larger number of clusters as the optimal model.

Table 3.2 Automatic Cluster Determination

	Mean of L	Mean of a	Mean of b	Mean of Anisotropy	Mean of Polarity	Mean of Contrast	Var of L	Var of a	Var of b	Var of Anisotropy	Var of Polarity	Varof Contrast	Region	Caption of Region	Image
Region 1	0.601	0.457	0.363	0.01	0.003	0.014	0.04	0.016	0.027	0.022	0.009	0.029	172000_1	sea	172000
Region 2	0.681	0.496	0.428	0.097	0.044	0.142	0.143	0.086	0.116	0.059	0.041	0.079	172000_2	sailing	172000
Region 3	0.666	0.512	0.415	0.095	0.043	0.137	0.133	0.091	0.09	0.05	0.044	0.055	172001_1	sailing	172001
Region 4	0.681	0.473	0.469	0.019	0.005	0.027	0.085	0.04	0.137	0.032	0.012	0.044	172001_2	sky	172001
Region 5	0.681	0.386	0.335	0.001	0	0.001	0.037	0.008	0.027	0.007	0.005	0.009	172002_1	sky	172002
Region 6	0.681	0.453	0.457	0.093	0.044	0.126	0.127	0.029	0.047	0.044	0.038	0.05	172002_2	sailing	172002
Region 7	0.681	0.416	0.433	0.05	0.011	0.068	0.051	0.023	0.041	0.025	0.011	0.03	172002_3	sea	172002
Region 8	0.681	0.466	0.472	0	0	0	0.036	0.015	0.026	0.006	0.005	0.006	172002_4	sailing	172002
Region 9	0.681	0.36	0.322	0	0	0	0.044	0.021	0.041	0.006	0.005	0.007	172003_1	sky	172003
Region 10	0.681	0.429	0.476	0.092	0.037	0.138	0.131	0.061	0.092	0.044	0.031	0.045	172003_2	sailing	172003
Region 11	0.681	0.382	0.402	0.042	0.013	0.07	0.064	0.031	0.057	0.028	0.014	0.038	172003_3	sea	172003
Region m-2															
Region m-1															
Region m															

3.4 Graph-Based Annotation Model

Our proposed Query Sensitive Random Walk method is in fact, capable of attacking the general problem of finding correlations between arbitrary modalities. In auto-captioning, it finds correlations between image features and caption words. We represent images, their regions and their caption words as nodes in a weighted graph. Nodes are linked together by four types of weighted links. See Figure 3.4.

3.4.1 **Image-Region Link** (IRL)

IRL is a link between an image and one of its region nodes. The weight of the link is estimated by:

$$IRL = \frac{1}{G1} \tag{3.16}$$

where G1 is the number of regions belonging to that image.

3.4.2 **Image-Caption Term Link** (ICL)

ICL is a link between an image and a caption term of that image. The weight of the link is estimated by:

$$ICL = \frac{1}{G2} \tag{3.17}$$

where G2 is the number of caption terms describing that image.

3.4.3 **Region-Region Link** (RRL)

RRL is a link connecting a region to a region of another image. Each region is represented by a vector of twelve features: six for mean and six for variance. The six mean components are the L, a, b, ac, pc, and c coordinates found after spatial averaging using TwoStep cluster at a selected scale; the anisotropy and polarity are each modulated by the contrast since they are meaningless in regions of low contrast. The six variance components are those of L, a, b, ac, pc, and c, access pixels of the region, i.e., $\left\langle \bar{L}, \bar{a}, \bar{b}, \overline{ac}, \overline{pc}, \bar{c}, \mathrm{var}(L), \mathrm{var}(a), \mathrm{var}(b), \mathrm{var}(ac), \mathrm{var}(pc), \mathrm{var}(c) \right\rangle$. Each region will have connections to k other regions that belong to other images. The weight of the link is calculated by 1- normalized Euclidean distance between the two region vectors whose value is greater than a threshold.

$$RRL = 1 - \sqrt{\sum (x-y)^2} \tag{3.18}$$

3.4.4 **Term Collocation Link** (TCL)

CTL is a link between two caption term nodes. Its weight is the collocation strength between two caption term vectors. The strength measures internal cohesion for determining a word which can be calculated as:

$$\text{Collocation } (x| y) = -p(x, y) \log p(y \mid x) \tag{3.19}$$

From this definition, the conditional probability $p(y|x)$ is easily obtained as follows:

$$p(y \mid x) = \frac{p(x \cap y)}{p(x)} \tag{3.20}$$

We need to decide on a threshold for the "closeness". There are many ways, but we decided to make the threshold adaptive: for each feature-vector, choose its k nearest neighbors, and associate them by connecting them with edges. Therefore, the edges added to relate similar regions are called the "nearest-neighbor" links (NN-links). We discuss the choice of k later, as well as the sensitivity of our results to k .

This model begins with graph-based features preparation for annotation process by collecting several features introduced in previous chapter. The graph-based features are passed into a model of Query Sensitive Random Walk to select a set of terms for annotation.

3.5 Word Ranking Model

3.5.1 Graph Random Walks

There are several methods similar to Random Walk with Restarts (RWR), including electricity based method (Zhu, Ghahramani, and Lafferty, 2003: 912-919), graph-based Semi-supervised learning (Faloutsos, McCurley and Tomkins, 2004: 118-127; Zhou, Bousquet, Lal, Weston and Scholkopf, 2003: 321-328) and so on. Exact solution of these methods usually requires the inversion of a matrix which is often diagonal dominant and of big size. Other methods sharing this requirement include regularized regression, Gaussian process regression (Rasmusen and Williams, 2006: 13-18), and so on. Other methods for RWR include Hub-vector decomposition based (Jeh and Widom, 2003); block structure based (Kamvar, Haveliwala, Manning and Golub, 2003; Sun, Qu, Chakrabarti and Faloutsos, 2005: 418-425); fingerprint based (Fogaras and Racz, 2004: 105-117), and so on. Many applications take random walk and related methods as the building block, including PageRank (Page, Brin, Motwani and Winograd, 1998), personalized PageRank (Haveliwala, 2002: 517-526), SimRank (Jeh and Widom, 2002: 538-543), neighborhood formulation in bipartite graphs (Sun, Qu, Chakrabarti and Faloutsos, 2005: 418-425), content-based image retrieval (He, Li, Zhang, Tong and Zhang, 2004: 9-16), cross modal correlation discovery (Pan, Yang, Faloutsos and Duygulu, 2004: 653-658), the BANKS system

(Aditya, Bhalotia, Chakrabarti, Hulgeri, Nakhe and Parag, 2002: 1083-1086), ObjectRank (Balmin, Hristidis and Papakonstantinou, 2004: 564-575), RalationalRank (Geerts, Mannila and Terzi, 2004: 552-563).

3.5.2 Random Walk with Restarts

Random Walk with Restarts (RWR) (Pan, Yang, Faloutsos and Duygulu, 2004) for uncaptioned keyword, they want to rank the caption nodes with respect to the user's image node, so we set the user's image node as a restart node. RWR operates as follows. For each node, RWR walker chooses randomly and moves to the next node among the available links with the exception to return to the restart node with probability c. To estimate the affinity of caption nodes with respect to image nodes, we have to find $r_{q(t)}$, the steady-state probability that the random walker will reach caption nodes from image nodes. The caption nodes, which are in the defined rank, are assigned as the captions of the query image. The defined rank is upon number of regions that cluster pixel in each image.

We are interested in the efficient precomputation of multiple BWR vectors. The algorithm for performing a single iteration is as follows:

1. Assign zero to all elements in $\vec{v}_q$ except the element which corresponds to the node of the query image I_q .

2. Let A be an adjacency matrix of image graph and column-normalized.

3. Initialize $\vec{u}_q = \vec{v}_k$.

4. The following recurrence is evaluated until $\vec{u}_q$ converges.

$$\overrightarrow{u_q} = (1-c)A\overrightarrow{u_q} + c\overrightarrow{v_q} \qquad (3.21)$$

3.5.3 Query Sensitive Random Walk

The image annotation problem is mapped to a graph ranking problem on the annotation graph constructed in the previous sections. QSRW (Query Sensitive Random Walk) is employed as the graph ranking algorithm for this phase.

The QSRW computes scores for each node by considering a random walker. If the random walk is at the query image node, it may choose to follow any available edge; however, if it is at other nodes, it performs a jump to the query image node with probability (1-α) before making a choice to follow an edge.

Our proposed approach is modified from Random Walk with Restarts (Pan et al., 2004) to estimate the affinity of image nodes to caption nodes. We use QSRW algorithm because it has ability to bias toward the restart node. The algorithm for performing a single iteration is as follows:

1. Assign zero of all elements in $\vec{v}_k$ except the element which corresponds to the node of the query image I_q .

2. Let W be an adjacency matrix of image graph and column-normalized.

3. Initialized $\vec{r}_i = \vec{v}_k$.

4. The following recurrence is evaluated until $\vec{r}_i$ converges.

$$r^{(i)}(p) = \begin{cases} \alpha \cdot \sum_{q=1}^{N} \frac{r^{(i-1)}(q) * d_{in}(q)}{d_{out}(q)} & \text{p is the query image node} \\ (1-\alpha)\frac{R^{(0)}}{s} + \alpha \cdot \sum_{q=1}^{N} \frac{r^{(i-1)}(q) * d_{in}(q)}{d_{out}(q)} & \text{otherwise} \end{cases} \qquad (3.22)$$

where

$d_{in}(q)$ = the (column-normalized) adjacency matrix,

$d_{out}(q)$ = a sum of weight matrix,

$R^{(0)}$ = a sum of the query image node,

s = a number of total nodes.

Once the algorithm reaches a steady state, only scores of term nodes are arranged in a descending order based on the node scored. The top n terms are selected as annotating terms for the query image where n can be the number of regions in the image or pre-specified by users.

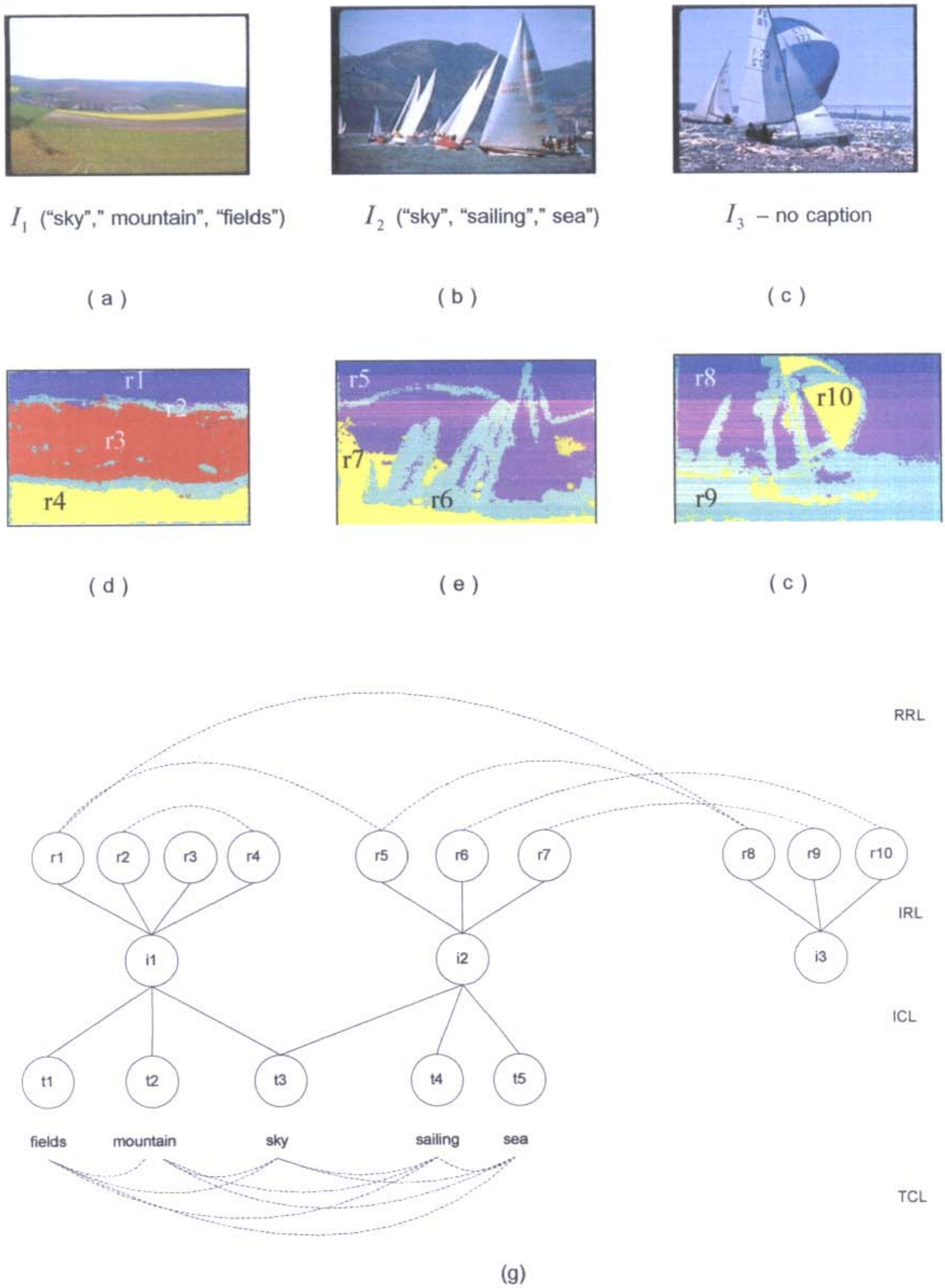


Figure 3.4 Image Annotation Graph

CHAPTER 4

EXPERIMENTAL EVALUTION

The Chapter 3, we introduced a new algorithm for image annotation model. In this chapter, we conduct the performance evaluation of the algorithm and present the experimental results. This chapter describes about selection of test data sets, standard measurement for performance evaluation, experimental results, and other results. Algorithms were implemented in the Java programming language and Matlab^{TM} and were run on a Windows XP platform.

4.1. Data Set Collection

In our experiments, we use 7 image data sets from Corel (Carson, Belongie, Greenspan and Malik, 2002: 1026-1038; Pan, Yang, Faloutsos and Duygulu, 2004), which are commonly used in previous works. On average, each data set has 2,110 regions, 700 images, and 36 captioning vocabularies. The resulting Image Annotation graph has 2,846 nodes.

The first effect does not hamper query performance; over segmenting uniform back ground regions just yields two blobs with similar color and texture descriptors, and the shape of background blobs is not important. Missing the true object boundary simply perturbs the region's color and texture descriptors slightly. Missing hard-to-find objects may cause those images not to be ranked highly in a query; however, users are generally looking for good examples of an easily visible object, so the practical effect on query performance is minimal. The last error truly hampers performance, because it causes good images to be missed in the query. Fortunately, this error occurs only rarely, less than 0.1 percent of the time, with smaller objects more likely to be missed than larger ones.

4.2. Performance Measures

For each test image, we compute the captioning accuracy as the percentage of caption terms which are correctly predicted. For a test image which has p correct caption terms, Image Annotation Graph will predict also p terms. If l terms are correctly predicted, then the captioning accuracy for this test image is defined as $\frac{l}{p}$.

Figure 4.1 shows some examples of the annotations given by QSRW (Query Sensitive Random Walk). Note that the QSWR graph used for this experiment is not the one shown in Figure 3.5, which is for illustration only. In Figure 4.1, QSRW gets the annotation correctly in Figue 4.1(a) and gets the word "sailing" and "sea" correctly (b); however, it mixes up hand-made objects ("hedge") with "straw" (c).

Figure 4.2(a) shows the captioning accuracy for the 7 data sets. We compare our results (with α = 0.5, k = 10 and with α = 0.66, k = 3) with the results reported in Random Walk with Restarts (RWR).

Query			
Actual Caption Terms	sailing, sea, sky	sailing, mountain, sea, sky	field, sky, barn, straw
QSWR	sea, sailing, sky	sailing, life-saving, sea, birds	hedge, straw barn, snow
	(a)	(b)	(c)

Figure 4.1 Example of captions generated by Image Annotation Graph. The predicted caption terms by QSWR are sorted by the estimated affinity values to the query image.

We also compare the captioning accuracy with even more recent machine vision methods: Random Walk with Restarts (RWR), and PageRank. The reported results of RWR and PageRank are based on the same data set as we used here.

Figure 4.2(b) compares the best average captioning accuracy among the 7 data sets reported by the RWR and PageRank, along with that of QSRW (with $\alpha = 0.5, k = 10$ and with $\alpha = 0.66, k = 3$). Although both RWR and PageRank are improvements over the TwoSteps Cluster algorithm, our generic QSRW approach outperforms the two algorithms (56.96% accuracy, versus 55.5% and 18.25%).

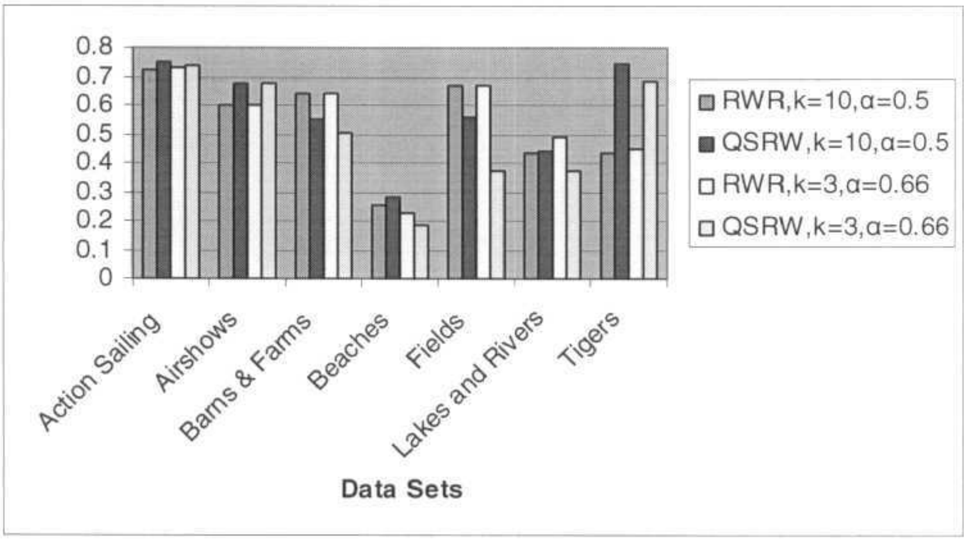
4.3. Effects of Model Parameters

We also performed experiment to find out how different values of the parameters α and k affect the captioning accuracy.

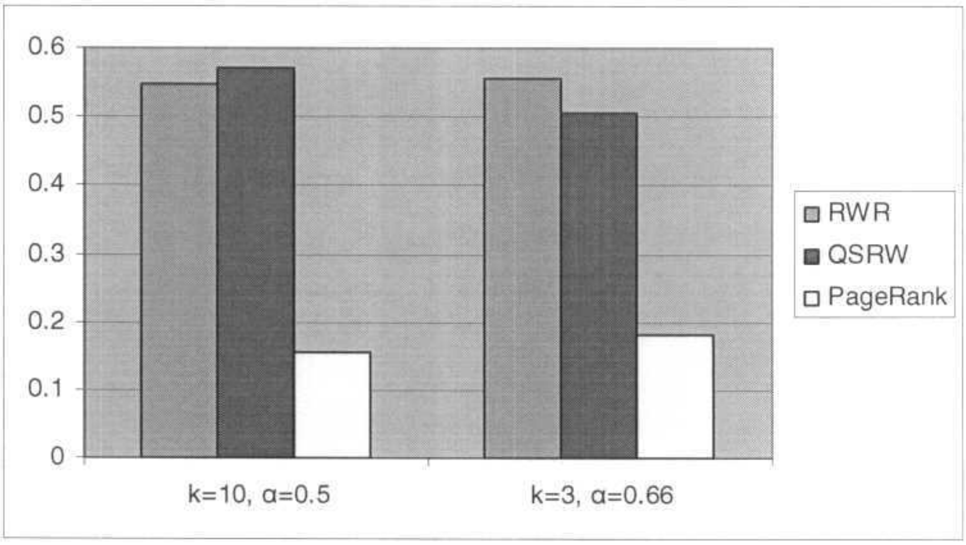
Figure 4.3 shows the captioning accuracy of Query Sensitive Random Walk using different values of restart probability α. The parameter k is fixed at 3, 5, 10, 30 and 50. The accuracy reaches a plateau as α grows from 0.4 to 0.9, which indicates the proposed Query Sensitive Random Walk algorithm is insensitive to the choice of α and k.

4.4 Other Results

Figure 4.4 shows the captioning accuracy of Query Sensitive Random Walk using different Image Annotation Graph between with Collocation and without Collocation. In all cases, Query Sensitive Random Walk used $\alpha = 0.5, k = 10$. Accuracy rate of Image Annotation Graph with Collocation is 56.96% where without Collocation is 54.56%.



a) Scores for RWR and QSRW by Group Data Sets



(b) Scores for RWR, PageRank and QSRW All Cases

Figure 4.2 Comparisons QSRW, RWR and PageRank (QSRW uses $\alpha = 0.5$, $k = 10$ and $\alpha = 0.66$, $k = 3$).

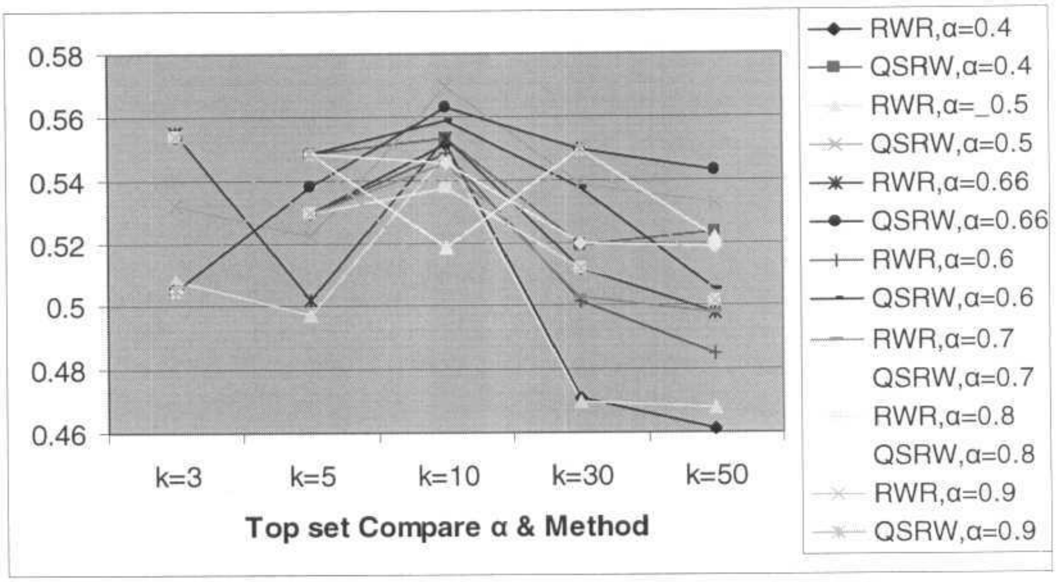


Figure 4.3 Varying the decay factor α and k shows the captioning accuracy of QSRW on data set using different values of k .

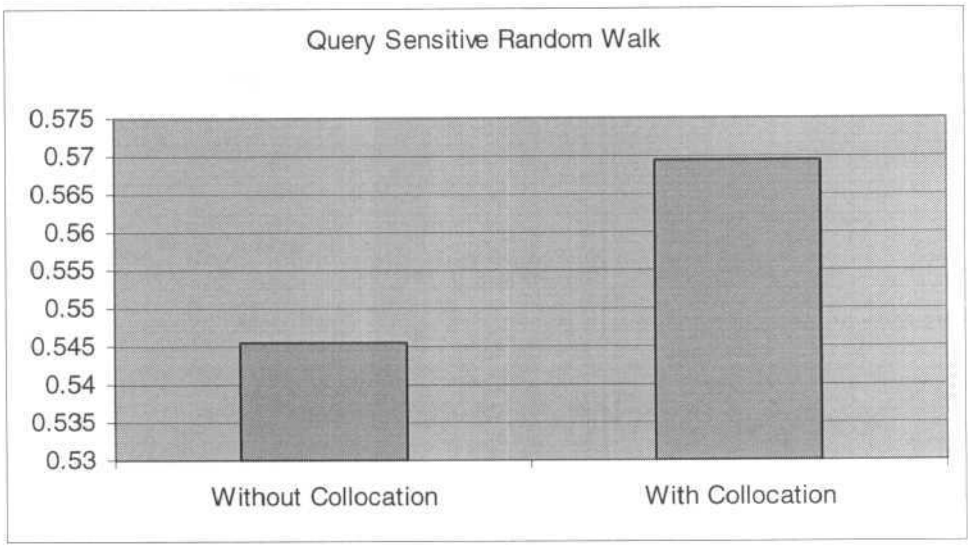


Figure 4.4 Comparisons QSRW between with Collocation and without Collocation. In all cases, QSRW used $\alpha = 0.5$, $k = 10$.

CHAPTER 5

CONCLUSIONS

Collections of images have been growing at a fast rate, from stock photo collections and proprietary databases to collections on the World Wide Web. Automatic image annotation is a process of labeling images with appropriate terms by learning associations between images' features and their caption terms.

This research proposes an image annotation approach based on a novel graph structure which includes similarities among regions within images and caption term collocation information, and caption terms are selected using Query Sensitive Random Walk, a graph random walk method biased toward the query image. A series of experiments are performed on a standard image data set to evaluate the proposed approach. The results show the annotation accuracy of 56.96%.

APPENDIX

Example Results

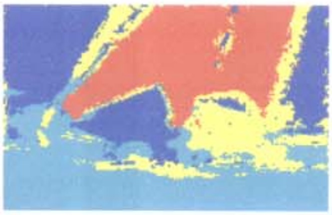
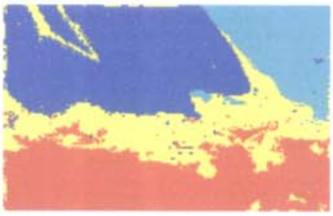
Image	Regions	Key words	QSRW	Accuracy (%)
		sea, sailing	sea, sailing	100
		sea, sailing, sky	sea, sailing, sky	100
		sailing, sky	sailing, sky	100
		sea, sailing, sky	sea, sailing, airplane	67.33
		sailing, sky	sailing, airplane	50

Figure 1 example of the sailing data set

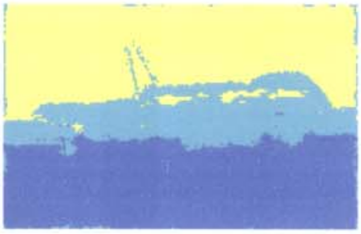
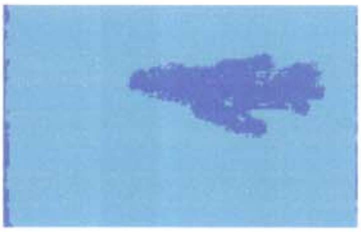
Image	Regions	Key words	QSRW	Accuracy (%)
		sky, balloon	sky, airplane	50
		airplane, sky, grass	airplane , airport, grass	67.33
		airplane, sky	airplane , sky	100
		airplane, sky	airplane , sky	100
		airplane, sky	airplane , sky	100

Figure 2 example of the air show data set

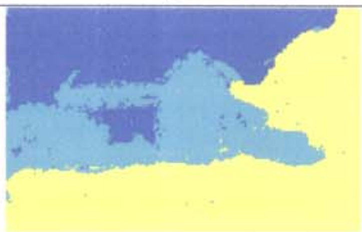
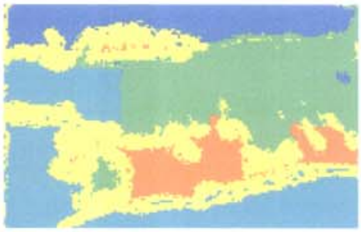
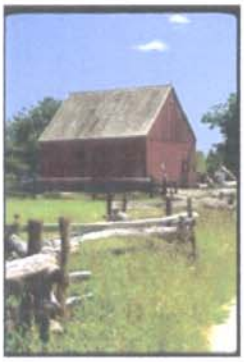
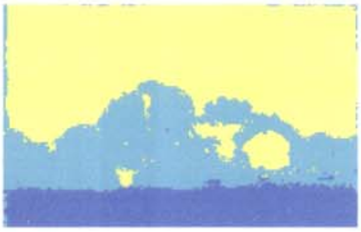
Image	Regions	Key words	QSRW	Accuracy (%)
		sky, barn, field	sky, barn, tree	67.33
		sky, barn, field, hedge	sky, barn, field, tree	75
		sky, barn, field	sky, barn, field	100
		sky, field	sky, field	100
		sky, barn, field	sky, barn, field	100

Figure 3 example of the barns and farms data set

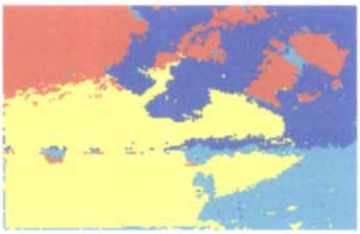
Image	Regions	Key words	QSRW	Accuracy (%)
		sky, stone, sea, grass	sky, barn, field, sea	50
		island, sky	island, sea	50
		sea, human, beach	sea, human, sky	67.33
		sea, stone, sky, tree	sea, stone, forest, sky	75
		sea, stone	sea, sky	50

Figure 4 example of the beaches data set

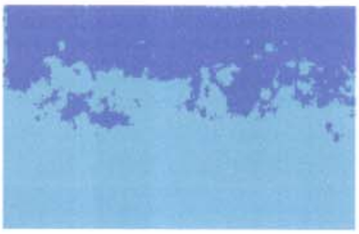
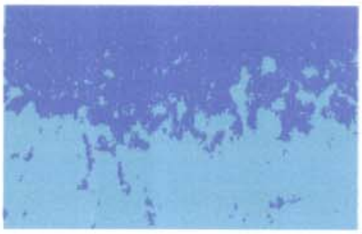
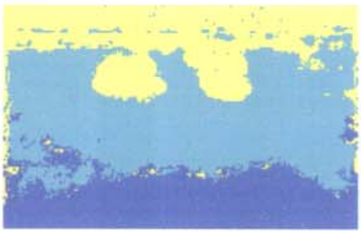
Image	Regions	Key words	QSRW	Accuracy (%)
		sky, field, mountain	sky, field, mountain	100
		field	field	100
		field	field	100
		field, horse	field, horse	50
		field, house	field, sky	50

Figure 5 example of the field data set

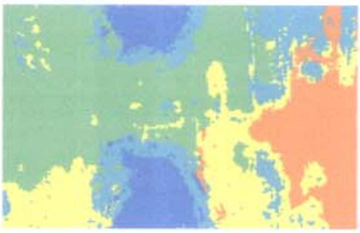
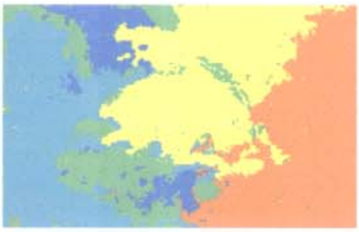
Image	Regions	Key words	QSRW	Accuracy (%)
		sky, lakes, mountain	sky, lakes, mountain	100
		river, forest	river, forest	100
		sky, lakes, field, mountain	sky, lakes, field, sea	75
		river, bridge	river, forest	50
		forest, sky	tree, sky	50

Figure 6 example of the lakes and rivers data set

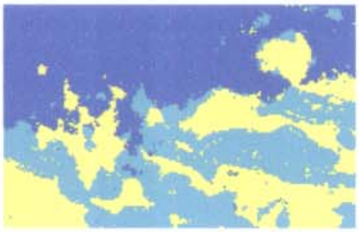
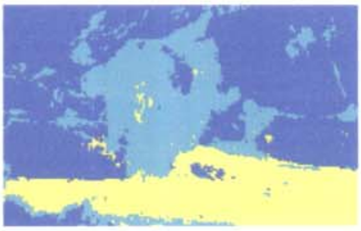
Image	Regions	Key words	QSRW	Accuracy (%)
		tiger, forest	tiger, forest	100
		sand, tiger, grass	sand, tiger, grass	100
		tiger, forest	tiger, grass	50
		tiger, forest	tiger, forest	100
		tiger, forest, river	tiger, forest, river	100

Figure 7 example of the tiger data set

BIBLIOGRAPHY

Aditya, B.; Bhalotia, G.; Chakrabarti, S.; Hulgeri, A.; Nakhe, C. and Parag, S. S. 2002. Banks: Browsing and Keyword Searching in Relational Databases. In **Proceeding of 28th International Conference on Very Large Data Bases**. California : Morgan Kaufmann. Pp. 1083–1086.

Balmin, A.; Hristidis, V. and Papakonstantinou, Y. 2004. Objectrank: Authority-Based Keyword Search in Databases. In **Proceeding of the 30th International Conference on Very Large Data Bases**. California : Morgan Kaufmann. Pp. 564–575.

Barnard, K. and Forsyth, D. A. 2001. Learning the Semantics of Words and Pictures. In **Proceedings of the 8th IEEE International Conference on Computer Vision**. Vancouver : IEEE. Pp. 408–415.

Barnard, K.; Duygulu, P.; Freitas, N.D.; Forsyth, D. A.; Blei, D. and Jordan, M. 2003. Matching Words and Pictures. **Journal of Machine Learning Research**. 3(February) : 1107–1135.

Carbonetto, P.; Freitas, N.D. and Barnard, K. 2004. A Statistical Model for General Contextual Object Recognition. In **Proceedings of the 8th European Conference Computer Vision**, Prague : Springer. Pp. 350-362.

Carson, C.; Belongie, S.; Greenspan, H. and Malik, J. 2002. Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying. **IEEE Transactions on Pattern Analysis and Machine Intelligence**. 24(8) : 1026-1038.

Chang, Y.C. and Lee, D.J. 2007. Color Image Quantization Using Color Variation Measure. In **Proceedings of the First IEEE Symposium on Computational Intelligence in Image and Signal Processing**. Hawaii : IEEE. Retrieved January 18, 2007 from http://www.ece.byu.edu/faculty/djlee/Publications/CIISP_ColorQ_042007.

Chen, Y.; Gan, Q. and Suel, T. 2002. I/O-efficient Techniques for Computing Pagerank. In **Proceeding of the 11th International Conference on**

Information and Knowledge Management. Virginia : ACM Pp. 549-557.

Clementine®. 2003. **Clementine® 8.0 Algorithms Guide**. Chicago : SPSS.

Faloutsos, C.; McCurley, K. S. and Tomkins, A. 2004. Fast Discovery of Connection Sub Graphs. In **Proceeding of The 10th ACM SIGKDD Conference on Knowledge Discovery and Data Mining**. Washington : ACM. Pp. 118–127.

Fogaras, D. and Racz, B. 2004. Towards Scaling Fully Personalized Pagerank. In **Proceeding of the Web-Graph: Third International Workshop**. Rome : Springer. Pp. 105–117.

Freeman, W.T. and Adelson, E.H. 1991. The Design and Use of Steerable Filters. **IEEE Transactions on Pattern Analysis and Machine Intelligence**. 13(9) : 891-906.

GaËrding, J. and Lindeberg, T. 1996. Direct Computation of Shape Cues Using Scale-Adapted Spatial Derivative Operators. **International Journal of Computer Vision**. 17(2) : 163-191.

Geerts, F.; Mannila, H. and Terzi, E. 2004. Relational Link-Based Ranking. In **Proceeding of 30th International Conference on Very Large Data Bases**. Toronto : VLDB. Pp. 552–563.

Granlund, G.H. and Knutsson, H. 1995. **Signal Processing for Computer Vision**. Dordrecht: Kluwer Academic Publishers.

Haveliwala, T.H. 2002. Topic-sensitive Pagerank In **Proceedings of the 11th International World Wide Web Conference**. New York : ACM. Pp. 517-526.

He, J.; Li, M.; Zhang, H.; Tong, H.; and Zhang, C. 2004. Manifoldranking Based Image Retrieval. In **Proceeding of ACM Multimedia**. New York : ACM. Pp. 9–16.

Jeh, G. and Widom, J. 2002. Simrank: A Measure of Structural Context Similarity. In **Proceeding of The 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.** Alberta : ACM. Pp. 538-543.

Jeh, G. and Widom, J. 2003. **Scaling personalized web search**. Retrieved January 12, 2007 from http://www2003.org/cdrom/papers/refereed/p185/html/ p185-jeh.html.

Kamvar, S.; Haveliwala, T.; Manning, C. and Golub, G. 2003. Exploiting the Block Structure of the Web for Computing Pagerank. **Stanford University Technical Report**. Retrieved December 8, 2006 from http://www.stanford.edu/~sdkamvar/papers/blockrank.pdf.

Kleinberg, J.M. 1999. Authoritative Sources in a Hyperlinked Environment. **Journal of the 9th ACM-SIAM Symposium on Discrete Algorithms**. 46(5) : 604-632.

Li J. and Wang, J. Z. 2003. Automatic Linguistic Indexing of Pictures by A Statistical Modeling Approach. In **Proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence**. PA : Dept. of Stat., Pennsylvania State University. 25(9) :1075-1088.

Maron, O. and Ratan, A.L. 1998. Multiple-instance Learning for Natural Scene Classification. In **Proceedings of the Fifteenth International Conference on Machine Learning**. California : Morgan Kaufmann. Pp. 341-349.

Page, L.; Brin, S.; Motwani, R. and Winograd, T. 1998. The PageRank citation ranking: Bringing order to the web. **Technical report, Stanford Digital Library Technologies Project**. Retrieved December 13, 2006 from http://www.cs.odu.edu/~mln/teaching/cs791s07/?method=getElement&element=~week4~KleinVuppala.pdf

Pan, J.-Y.; Yang, H.-J.; Faloutsos, C. and Duygulu, P. 2004. Automatic Multimedia Cross-model Correlation Discovery. In **Proceeding of the 10th ACM SIGKDD conference on Knowledge Discovery and Data Mining**. Washington : ACM. 653–658.

Pan, J-Y.; Yang, H.-J.; Faloutsos, C. and Duygulu, P. 2004. GCap: Graph-based Automatic Image Captioning. In **Proceedings of Conjunction with Computer Vision Pattern Recognition Conference (CVPR 04)**. Washington, USA. Retrieved June 12, 2006 from http://www.db.cs.cmu.edu/db-site/Pubs/Lib/mdde04gcap/

MDDE04GCap.pdf .

Pan, J-Y.; Yang, H.-J.; Faloutsos, C. and Duygulu, P. 2004. GCap: Graph-based Automatic Image Captioning. In **Proceedings of the 4th International Workshop on Multimedia Data and Document Engineering (MDDE 04)**. Washington, USA. Retrieved June 12, 2006 from http://www.db.cs.cmu.edu/db-site/Pubs/Lib/mdde04gcap/ MDDE04GCap.pdf .

Rasmusen, C. E. and Williams, C. 2006. **Gaussian Processes for Machine Learning**. MA: MIT Press.

Sun, J.; Qu, H.; Chakrabarti, D. and Faloutsos, C. 2005. Neighborhood Formation and Anomaly Detection in Bipartite Graphs. In **Proceeding of the 5th IEEE International Conference on Data Mining**. Texas : IEEE. Pp. 418–425.

Wenyin, L.; Dumais, S.; Sun, Y.; Zhang, H.; Czerwinski, M. and Field, B. 2001. Semi Automatic Image Annotation. In **Proceedings of INTERACT2001, 8th IFIP TC.13 Conference on Human-Computer Interaction**. Tokyo, Japan. Pp. 326-333.

Wyszecki, G. and Styles, W. S . 1982. **Color Science: Concepts and Methods.** New York : Wiley.

Zhang, T.; Ramakrishnon, R. and Livny, M.. 1996. BIRCH: An Efficient Data Clustering Method for Very Large Databases. In **Proceedings of the ACM SIGMOD Conference on Management of Data**. Montreal : ACM. Pp. 103-114.

Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J. and Scholkopf, B. 2003. Learning with Local and Global Consistency. In **Proceeding of The 17th Annual Conference on Neural Information Processing Systems.** Whistler Pp. 321-328.

Zhu, X.; Ghahramani, Z. and Lafferty J. D. 2003. Semi-supervised Learning Using Gaussian Field and Harmonic Functions. In **Proceeding of The 12th International Conference on Machine Learning**. Washington, USA. Pp. 912–919.

BIOGRAPHY

NAME	Mr. Krittapad Suriya
ACADEMIC BACKGROUND	Bachelor of Science in Statistics from Srinakharinwirot University, Bangkok, Thailand in 2002.
PRESENT POSITION	System Analyst, Pro Technical Info Services Co., Ltd.