ImageSeer: Clustering and Searching WWW Images Using Link and Page Layout Analysis

Xiaofei He
Deng Cai
Ji-Rong Wen
Wei-Ying Ma
Hong-Jiang Zhang

April 1, 2004

Technical Report
MSR-TR-2004-38

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

ImageSeer: Clustering and Searching WWW Images Using Link and Page Layout Analysis

Xiaofei He[1*], Deng Cai[*], Ji-Rong Wen[*], Wei-Ying Ma[*] and Hong-Jiang Zhang[*]

[*]Microsoft Research Asia
Beijing, China
{jrwen, wyma, hjzhang}@microsoft.com
cai_deng@yahoo.com

[1]Department of Computer Science
University of Chicago
xiaofei@cs.uchicago.edu

ABSTRACT

Due to the rapid growth of the number of digital images on the Web, there is an increasing demand for effective and efficient method for organizing and retrieving the images available. This paper describes *ImageSeer*, a system for clustering and searching WWW images. By using a vision-based page segmentation algorithm, a web page is partitioned into blocks, and the textual and link information of an image can be accurately extracted within the block containing that image. The textual information is used for image representation. By extracting the page-to-block, block-to-image, block-to-page relationships through link structure and page layout analysis, we construct an image graph. Our method is less sensitive to noisy links than previous methods like PicASHOW, and hence the image graph can better reflect the semantic relationship between images. With the graph models, we use techniques from spectral graph theory and Markov Chain theory for image ranking, clustering and embedding. Some experimental results are given in the paper.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Clustering, Query formulation, Retrieval Models.* I.4.10 [**Image Representation**]: Image Processing and Computer Vision – *Multidimensional, Statistical.*

General Terms

Algorithms, Management, Design, Theory

Keywords

Web image search, web page segmentation, spectral analysis, ImageRank, image clustering, graph model, link analysis

1. INTRODUCTION

The emergence of World Wide Web has posed many practical problems. Among them, how to find the information that people wants is needlessly one of the most important. In this paper, we consider the problem of clustering and searching images on the web.

Traditional image retrieval has attracted the attention of many researchers. Basically, it falls into two main categories, content based image retrieval (CBIR) [16][19][22][25] and semantics based image retrieval [12][13][31][32]. Both of these two techniques are based on *small* and *static* (compared to the Internet) image databases, like family albums. These limitations make traditional image retrieval techniques not directly applicable to web image search. Also, different from traditional image retrieval, there is a lot of additional information on the web, such as surrounding texts and hyperlinks, which is useful to enhance image search performance.

This paper describes a web image search engine called *ImageSeer*. Like text search engines, ImageSeer does not have to access the original data to respond to a query; all analysis of the image and surrounding text is done off-line during the creation of the database. In this way, ImageSeer can give fast query responses to a possibly huge number of users. To make it practically applicable in terms of scalability and performance, we especially focused on three fundamental problems, *i.e. representation, similarity measure and ranking*. Technically speaking, ImageSeer also contains three parts, *VIsion-based Page Segmentation (VIPS)* [5][6][30], *link and page layout based graph model, and spectral analysis for image ranking, clustering, and embedding*.

Most of previous web-based applications [4][15][16][36][37] regard web pages as information units. However, it is the case that a single web page often contains multiple semantics. Thus, from the perspective of semantics, a web page should not be the smallest unit. For example, the page at http://news.yahoo.com/ contains multiple news topics, such as business, entertainment, sport, technology, etc. The hyperlinks contained in different semantic blocks usually point to the pages of different topics. Naturally, it is more reasonable to regard the semantic blocks as the smallest units of information. By using our VIPS algorithm, each page can be segmented into several semantic blocks. For image search, we are interested in those blocks containing images (called *image blocks*). The surrounding texts extracted within the image block are used to index the image.

Besides textual information, link structure is also of great importance. Textual information reflects the semantics of a single image, while link structure reflects the semantic relationships between images. Most of previous link analysis algorithms such as HITS [15] and PageRank [4] use link structure to construct a page-to-page graph. In other words, a link is from page to page. In our work, we consider three kinds of relationships, *i.e.* block-to-page (link structure), page-to-block (page layout), and block-to-image (inclusion relation), which ultimately results in three graphs, *i.e.* page-to-page graph, block-to-block graph and image-to-image graph. It is worthwhile to note that, in our framework, a link is from block to page rather than from page to page.

With the graph models, we use techniques from spectral graph theory [4][11][19] and Markov Chain theory for image ranking, clustering and embedding. By embedding we mean that

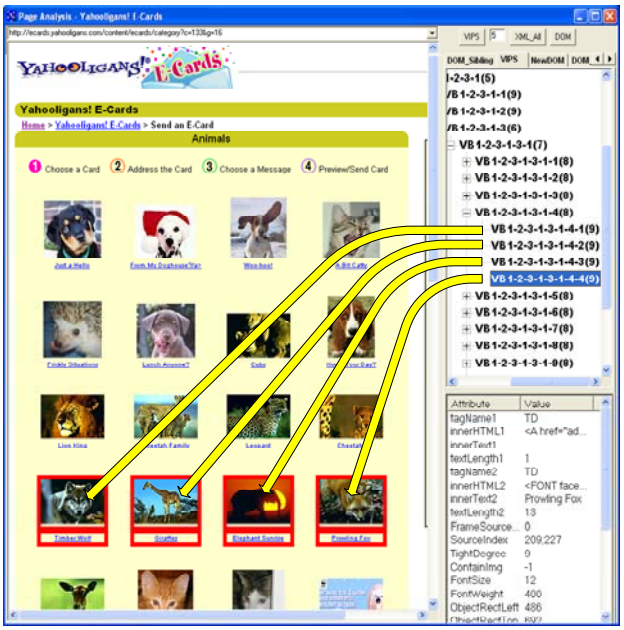

Figure 1. The interface of our VIPS page segmentation system. The surrounding texts are extracted within the image blocks (with red frame).

each image can be endowed with a vector representation. A Markov Chain can be induced from the image-to-image graph. The stationary distribution of the Markov Chain gives a quantitative evaluation of the importance of each image. Thus the images can be ranked according to their importance. Also, the images can be clustered into semantic classes by using spectral graph partitioning. Likewise, the image embedding problem is formulated as a graph embedding problem. The resulting vector representations of images can be used for browsing or 2D visualization purposes.

The rest of this paper is organized as follows. Section 2 relates a list of previous works to our work. In Section 3, we describe the VIPS page segmentation algorithm and its application to surrounding text extraction. In Section 4, we describe how to build the graph models. We present our methods for image ranking, clustering, and embedding in Section 5. Some experimental evaluations are provided in Section 6. Finally, we give concluding remarks and future work in Section 7.

2. PREVIOUS WORKS

Image search is a long standing research problem. Previous work on *web* image search mainly falls into two categories, content-based [10][24][33] and link-based [16]. Note that, we do not distinguish text-based search as another category simply because almost all practical search engines will use textual information.

For content-based web image search, the typical systems include WebSeer [10], ImageRover [24], WebSeek [33], etc. All of them combine the textual information and visual information (color, texture, shape, etc.) for image indexing. However, content based computation is very extensive, and hence it might not be practical for image search on the Internet.

The typical link-based image search system is PicASHOW [16]. The basic premise of PicASHOW is that a page *p* displays (or link to) an image when the author of *p* considers the image to be of value to the viewer of the page. Thus, PicASHOW first assembles a large collection of pages relevant to the query, and then the images contained in those pages are ranked according to several link structure analyzing algorithms. We are especially interested in PicASHOW since our method can also be classified into this category.

It is worthwhile to highlight several aspects of the proposed approach here:

1. In PicASHOW, there are two basic assumptions: (a) Images co-contained in **pages** are likely to be related to the same topic. (b) Images contained in pages that are co-cited by a certain **page** are likely related to the same topic. In fact, one can easily find many counter examples due to the fact that a page generally contains multiple different semantics. The images contained in different semantic blocks are likely related to different topics. The assumptions of our approach are the following: (a) Images co-contained in **blocks** are likely to be related to the same topic. (b) Images contained in pages that are co-cited by a certain **block** are likely related to the same topic.
2. PicASHOW applies link analysis algorithms to rank the images, which is computed on-line. The computation involved in our approach is off-line. Thus, our approach can provide faster response to user's query. This makes our approach much more practical.
3. The framework of analysis presented in this paper contains three technical parts, *i.e.* page segmentation, graph model and spectral analysis. Within this framework, discovering semantic and geometrical structure of the web image collections becomes possible (see [14] for details). Specifically, this framework provides a ranking scheme, as well as a clustering and embedding scheme for the web images. All of them together give an organization scheme for web images, which can be used for browsing purpose [14].
4. Although our primary interest in this paper is in image, our framework of analysis actually provides a way of block-level link analysis. For example, within this framework, an enhanced PageRank can be computed from the page-to-page graph which is induced from the block-level link structure analysis rather than traditional page-level analysis.

As a result of all these features, we expect the block based techniques to be a natural alternative to page based techniques in exploratory data analysis on the Web.

3. SURROUNDING TEXT EXTRACTION USING VIPS PAGE SEGMENTATION

The VIsion-based Page Segmentation (VIPS) algorithm [5][6] aims to extract the semantic structure of a web page based on its visual presentation. Such semantic structure is a tree structure; each node in the tree corresponds to a block. Each node will be assigned a value (*Degree of Coherence*) to indicate how coherent of the content in the block based on visual perception. The VIPS algorithm makes full use of page layout feature. It first extracts all the suitable blocks from the html DOM tree, and then it finds the separators between these blocks. Here, separators denote the horizontal or vertical lines in a web page that visually cross with no blocks. Based on these separators, the semantic tree of the web page is constructed. Thus, a web page can be represented as a set of blocks (leaf nodes of the semantic tree). For details, see [6].

Figure 2. The useful links are within the image blocks, while the noisy links are outside the image blocks.

Compared with DOM based methods, the segments obtained by VIPS are much more semantically aggregated. Noisy information, such as navigation, advertisement, and decoration can be easily removed because they are often placed in certain positions of a page. Contents with different topics are distinguished as separate blocks.

The VIPS algorithm can be naturally used for surrounding texts extraction. For each image, there is at least one (sometimes an image is cited repeatedly) image block that contains that image. Intuitively, the surrounding texts should be extracted within the image block. Figure 1 gives a simple example*. As can be seen, the surrounding texts are accurately identified. Besides surrounding texts, some other textual information is also useful for image representation, such as image file title, image ALT (alternate text), and page title.

Once we obtain text representations of the images, the web image search problem becomes a text information retrieval problem. Thus, we can apply the traditional text retrieval techniques, such as inverted indexing, TF-IDF weighting and cosine similarity measure, etc., for comparing the images to the query keywords. Unfortunately, due to the large amount of images available on the Web, many images end up having the same textual representations, indicating that textual representation alone is insufficient for image ranking. Therefore, we need to use another kind of information, i.e. link structure, to compute importance ranking. This motivates us to consider the WWW images as a graph.

* The URL of the presented web page is: http://ecards.yahooligans.com/content/ecards/category?c=133&g=16

4. GRAPH MODELS

The VIPS page segmentation algorithm does not only help to extract the meaningful surrounding texts, but also help to extract the useful links. In this section, we describe how to construct a block-to-block and image-to-image graphs. Like page-to-page graph model, the block-to-block model might be useful for many web based applications, such as web information retrieval and web page categorization, but in this paper our primary purpose is for image search and organization. Our graph model is induced from three kinds of relationships, *i.e.* ***block-to-page***, ***page-to-block***, and ***block-to-image***. We begin with some definitions. Let P denotes the set of all the web pages, $P = \{p_1, p_2, \ldots, p_k\}$, where k is the number of web pages. Let B denotes the set of all the blocks, $B = \{b_1, b_2, \ldots, b_n\}$, where n is the number of blocks. It is important to note that, for each block there is only page that contains that block. Let $I = \{I_1, I_2, \ldots, I_m\}$ denote the set of all the images on the web, where m is the number of the web images. $b_i \in p_j$ means the block i is contained in the page j. Similarly, $I_i \in b_j$ means the image i is contained in the block j.

4.1 Block-Based Link Structure Analysis

The block-to-page relationships are obtained from link analysis. Link analysis has proven to be very effective in web search [4][15][34]. However, a web page typically contains several semantic blocks. Different blocks have different meanings. Therefore, it might be more reasonable to consider the hyperlinks as from block to page, rather than from page to page. Let Z denote the block-to-page matrix with dimension $n \times k$. Z can be formally defined as follows:

$$Z_{ij} = \begin{cases} 1/s_i & \text{if there is a link from block } i \text{ to page } j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where s_i is the number of pages that block i links to. Z_{ij} can also be viewed as a probability of jumping from block i to page j. The block-to-page relationship gives a more accurate and robust representation of the link structures of the Web. For image search, those links outside the image blocks are regarded as noisy links, as shown in Figure 2. It is important to note that, traditional link-based image search method like PicASHOW does not distinguish between the noisy links and useful links. Some detailed comparison between our link structure analyzing method and PicASHOW's method is given in the experimental section.

Figure 3 shows an example of block-to-page link structure. As can be seen, as the noisy links are eliminated the resulting link structure of images is much more accurate. The outlinks in image blocks have very high probability to point to those pages containing the images related to the same topics. The block-based link structure extracted by our method is much more meaningful than that extracted by previous methods, such as PageRank, HITS, and PicASHOW, which do not distinguish the useful links from noisy links [4][15][16].

4.2 Page Layout Analysis

The page-to-block relationships are obtained from page layout analysis. Let X denote the page-to-block matrix with dimension $k \times n$. As we have described above, each web page can be

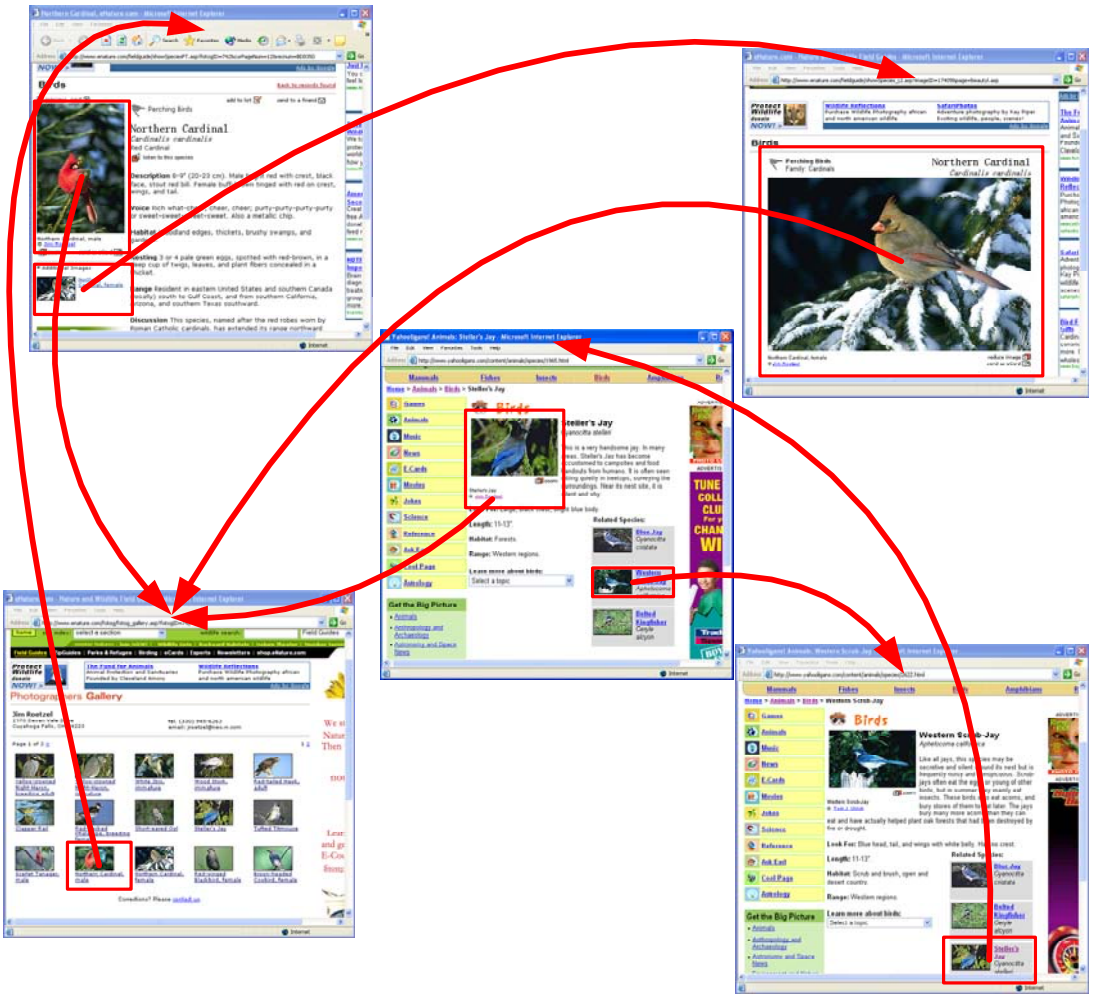

Figure 3. The block-to-page link structure. The red arrows denote the links from image block to web pages.

segmented into blocks. Thus, X can be naturally defined as follows:

$$X_{ij} = \begin{cases} 1/s_i & if\ b_j \in p_i \\ 0 & otherwise \end{cases} \quad (2)$$

where s_i is the number of blocks contained in page i. The above formula assigns equal importance value to each block in a page. It is simple but less practical. Intuitively, some blocks with big size and centered position are probably more important than those blocks with small size and margin position. This observation leads to the following formula,

$$X_{ij} = \begin{cases} f_{P_i}(b_j) & if\ b_j \in p_i \\ 0 & otherwise \end{cases} \quad (3)$$

where f is a function which assigns to every block b in page p an importance value. Specifically, the bigger $f_p(b)$ is, the more important the block b is. f is empirically defined below,

$$f_p(b) = \alpha \frac{\text{size of block } b \text{ in page } p}{\text{dist. from the center of } b \text{ to the center of screen}} \quad (4)$$

where α is a normalization factor to make the sum of $f_p(b)$ to be 1, *i.e.*

$$\sum_{b \in p} f_p(b) = 1 \quad (5)$$

Note that, $f_p(b)$ can also be viewed as a probability that the user is focused on the block b when viewing the page p.

Some more sophisticated definitions of f can be formulized by considering the background color, fonts, etc. Also, f can be learned from some pre-labeled data (the importance value of the blocks can be defined by people) as a regression problem by using learning algorithms such as SVM [28], neural networks, etc. We believe these methods will achieve a better result, yet it is beyond the scope of this paper and left for future study.

4.3 Block Analysis

Let Y denote the block-to-image matrix with dimension $n \times m$. For each image, there is at least one block that contains this image. Thus, Y can be simply defined below:

$$Y_{ij} = \begin{cases} 1/s_i & if\ I_j \in b_i \\ 0 & otherwise \end{cases} \quad (6)$$

where s_i is the number of images contained in the image block b_i. Also, for those image blocks, the surrounding texts of the images are extracted to represent the images, as described in Section 3.

4.4 Building Graph Models

In the last three subsections, we have constructed three affinity matrices, *i.e.* block-to-page, page-to-block, and block-to-image. Based on these three matrices, we can build three graph models, *i.e. page graph* $G_P(V_P, E_P, W_P)$, *block graph* $G_B(V_B, E_B, W_B)$, and *image graph* $G_I(V_I, E_I, W_I)$. For each graph, V is the set of the nodes (page, block, image, respectively), E is the set of edges linking two nodes, W is a weight matrix defined on the edges. We begin with the page graph.

4.4.1 Page Graph

When constructing a graph, we essentially define a weight matrix on the edges. W_P can be simply defined as follows. $W_P(i, j)$ is 1 if page i links to page j, and 0 otherwise. This definition is pretty simple yet has been widely used as the first step to many applications, such as PageRank [4], HITS [15], community mining [9], etc. However, based on our previous discussions, different blocks in a page have different importance. Therefore, those links in blocks with high importance value should be more important than those in blocks with low importance value. In other words, a user might prefer to follow those links in important blocks. This consideration leads to the following definition of W_P,

$$W_P(\alpha,\beta) = \sum_{b\in\alpha} f_\alpha(b) Z(b,\beta), \quad \alpha,\beta \in P \quad (7)$$

or

$$W_P = XZ \quad (8)$$

X is a $k \times n$ page-to-block matrix and Z is a $n \times k$ block-to-page matrix, thus W_P is a $k \times k$ page-to-page matrix.

Here we provide a simple analysis of our definition of W_P from the probabilistic viewpoint. Let's consider $W_P(\alpha, \beta)$ as a probability $Prob(\beta|\alpha)$ of jumping from page α to page β. Since page α is composed of a set of blocks, we have

$$Prob(\beta \mid \alpha) = \sum_{b\in\alpha} Prob(\beta \mid b)\, Prob(b \mid \alpha) \quad (9)$$

where $Prob(\beta|b)$ is actually $Z(b, \beta)$ and $Prob(b|\alpha)$ is $f_\alpha(b)$.

Finally, it would be interesting to see under what conditions our definition of W_P reduces to the ordinary definition. This occurs when the function $f(b)$ is defined as the number of links contained in block b.

4.4.2 Block Graph

The block graph is constructed over the blocks. Let's first consider a jump from block a to block b. Suppose a user is looking at block a. In order to jump to the block b, he first jumps to page β which contains block b, and then he focuses his attention on block b. Thus, a natural definition of W_B is as follows,

$$\begin{aligned} W_B(a,b) &= Prob(b \mid a) \\ &= \sum_{\gamma\in P} Prob(\gamma \mid a)\, Prob(b \mid \gamma) \\ &= Prob(\beta \mid a)\, Prob(b \mid \beta) \\ &= Z(a,\beta) X(\beta,b), \quad a,b \in B \end{aligned} \quad (10)$$

or

$$W_B = ZX \quad (11)$$

where W_B is a $n \times n$ matrix. By definition, W_B is clearly a probability transition matrix. However, there is still one limitation of this definition such that it is unable to reflect the relationships between the blocks in the same page. Two blocks are likely related to the same topics if they appear in the same page. This leads to a new definition,

$$W_B = (1-t)ZX + tDX^T X \quad (12)$$

where t is a suitable constant. D is a diagonal matrix, $D_{ii} = \sum_j (X^T X)_{ij}$. $(X^T X)_{ij}$ is zero if block i and block j are contained in two different web pages. It is easy to check that the sum of each row of $DX^T X$ is 1. Thus, W_B can be viewed as a probability transition matrix such that $W_B(a, b)$ is the probability jumping from block a to block b. Finally, it is worth noting that t is typically set to be very small, 0~0.1, since in many cases, different blocks are likely related to different topics even though they appear in the same page.

4.4.3 Image Graph

Once the block graph is obtained, the image graph can be constructed correspondingly by noticing the fact that every image is contained in at least one block. Let's consider the jump from image i to image j. From image i we first see the block α containing image i. By block graph, we get a jump from block α to block β containing image j. Finally, we stopped at image j. In this way, the weight matrix of the image graph can be defined as follows:

$$W_I(i, j) = \sum_{i\in\alpha,\, j\in\beta} W_B(\alpha, \beta) \quad (13)$$

or

$$W_I = Y^T W_B Y \quad (14)$$

where W_I is a $m \times m$ matrix. If two images i and j are in the same block, say b, then $W_I(i, j) = W_I(b, b) = 0$. However, the images in the same block are supposed to be semantically related. Thus, we get a new definition as follows:

$$W_I = tDY^T Y + (1-t)Y^T W_B Y \quad (15)$$

where t is a suitable constant and D is a diagonal matrix, $D_{ii} = \sum_j (Y^T Y)_{ij}$. Like W_B, W_I can be viewed as a probability transition matrix. t is typically set to be large, 0.7~0.9, since in many cases two images are related to the same topic if they appears in the same block.

5. SPECTRAL ANALYSIS OF THE GRAPH MODELS

The following section is based on the standard spectral graph theory and Markov Chain theory. See [7][11][19] for a comprehensive reference. Spectral techniques use information contained in the eigenvectors and eigenvalues of a data affinity (i.e., item-item similarity) matrix to detect structure. Such an approach has proven effective on many tasks, including information retrieval [8], web search [4], image segmentation [25], word class detection [3], face recognition [35] etc. In this section, we will describe the ImageRank algorithm using spectral techniques. By using spectral techniques on the graph models obtained previously, we can compute the ImageRank [14] of every image and cluster them into semantic groups.

5.1 ImageRank

As described in the introduction section, one of the fundamental problems in web image search is ranking. Ranking by text information alone is insufficient since some images can have the same textual representation. In this section, we describe ImageRank which gives every image an importance value. We expect that the text information combined with the importance value will give a better ranking scheme than each alone.

Let M denotes the random walk naturally induced from the image graph. By our definition, the weight matrix W of the graph is also the probability transition matrix of M. Now let us consider a random surfer on the graph. He jumps from image i to image j with probability W_{ij}. There is also a possibility that the surfer does not follow the probability transition matrix induced from the web structure but jumps to an image picked uniformly and at random from the collection. Thus, the new probability transition matrix P is defined as follows:

$$P = \varepsilon W + (1-\varepsilon)U \tag{16}$$

where ε a parameter, typically set to 0.1~0.2. In our experiments, it is 0.15. U is a transition matrix of uniform transition probabilities ($U_{ij} = 1/m$ for all i, j). In fact, the introduction of U makes the graph connected and hence the stationary distribution of the random walk always exists.

Imagine the random surfer keeps jumping and finally stops at image k in probability π_k. $\pi = (\pi_1, \pi_1, \ldots, \pi_m)$ is often called stationary distribution. It can be computed from the following equation:

$$P^T \pi = \pi \tag{17}$$

Clearly, π is an eigenvector of P^T with eigenvalue 1.

It is worth noticing that, in [14] the random walk was constructed in terms of visual features of the images. For web image search, this might not be practical because the image feature extraction and their similarity measures are computationally extensive.

5.2 Spectral Embedding and Clustering of the Images

Spectral graph embedding and clustering [1][7][11][19][21] connect to each other in spirit in the sense that both of them can reduce to similar eigenvector problem. Therefore, we will describe them together in a single section. Spectral graph embedding can be viewed as the first step to spectral clustering. Therefore, let us first consider image graph embedding. Here, by embedding we mean that each image is endowed with a vector representation in Euclidean space such that the distance between two images reflects their semantic similarity.

In Section 4, we have obtained a weight matrix of the image graph, W_I. We first convert it into a similarity matrix S such that $S = 1/2(W_I + W_I^T)$ which is symmetric. Note that content-based (color, texture, shape, etc.) similarity measure between images has been researched extensively in computer vision community in past decades [18][27]. The visual content of images might be helpful to define the optimal similarity measure. However, it almost impossible to apply it to web images because of scalability issue. The similarity measure we used is induced from the image graph, and hence it is much more computationally tractable.

Now, suppose y_i is an one-dimensional vector representation of image i. The optimal $\mathbf{y} = (y_1, \ldots, y_m)$ is obtained from the following objective function:

$$\min_{\mathbf{y}} \sum_{i,j} (y_i - y_j)^2 S_{ij} \tag{18}$$

The objective function with the choice of S_{ij} incurs a heavy penalty if semantically related images are mapped far apart. Therefore, minimizing it is an attempt to ensure that if image i and image j are semantically related then y_i and y_j are close to each other. Let D be a diagonal matrix whose i^{th} element is the row (or column, since S is symmetric) sum of S, $D_{ii} = \sum_j S_{ij}$. By simple algebra formulation, we have:

$$\begin{aligned}
&\frac{1}{2}\sum_{i,j}(y_i - y_j)^2 S_{ij} \\
&= \frac{1}{2}\sum_{i,j}(y_i^2 - 2y_i y_j + y_j^2) S_{ij} \\
&= \frac{1}{2}\left(\sum_{i,j} y_i^2 S_{ij} + \sum_{i,j} y_j^2 S_{ij} - 2\sum_{i,j} y_i y_j S_{ij}\right) \\
&= \frac{1}{2}\left(\sum_{i,j} y_i^2 S_{ij} + \sum_{i,j} y_j^2 S_{ij} - 2\sum_{i,j} y_i y_j S_{ij}\right) \\
&= \frac{1}{2}\left(2\mathbf{y}^T D\mathbf{y} - 2\mathbf{y}^T S\mathbf{y}\right) \\
&= \mathbf{y}^T L\mathbf{y}
\end{aligned} \tag{19}$$

where $L = D - S$. L is generally called Laplace matrix, or graph Laplacian. It is positive semi-definite. Thus, the minimization problem reduces to finding:

$$\min_{\mathbf{y}^T\mathbf{y}=1} \mathbf{y}^T L\mathbf{y} \tag{20}$$

The solution is given by minimum eigenvalue solution to the generalized eigenvalue problem:

$$L\mathbf{y} = \lambda\mathbf{y} \tag{21}$$

Let $(\mathbf{y}^0, \lambda^0), (\mathbf{y}^1, \lambda^1), \ldots, (\mathbf{y}^{m-1}, \lambda^{m-1})$ be the solutions to the above equation, and $\lambda^0 < \lambda^0 < \ldots < \lambda^{m-1}$. It is easy to check that $\lambda^0 = 0$ and $\mathbf{y}^0 = (1, 1, \ldots, 1)$. Therefore, we leave out the eigenvector $\mathbf{y}^0$

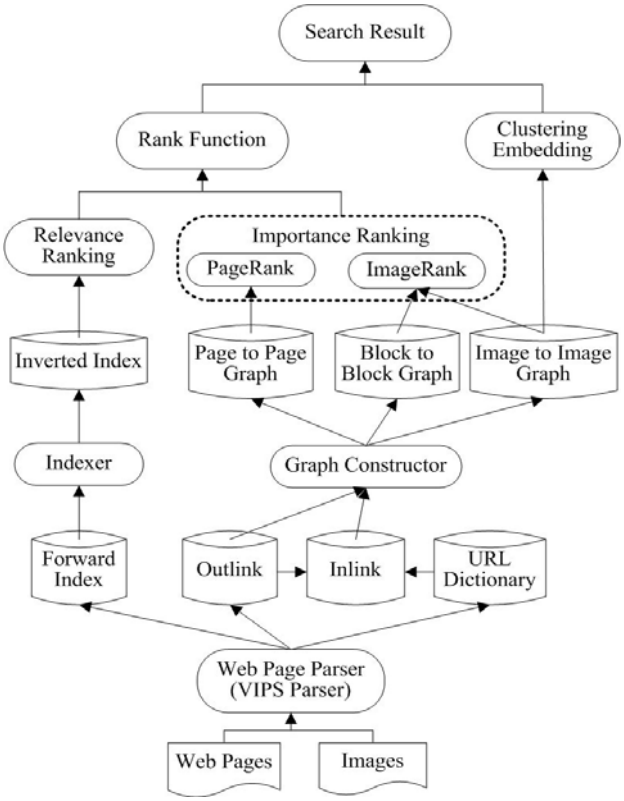

Figure 4. Design of our WWW image search system

and use the next k eigenvectors for embedding in k-dimensional Euclidean space:

$$image\ \ j \leftarrow (\mathbf{y}^1(j), \cdots, \mathbf{y}^k(j)) \qquad (22)$$

where $\mathbf{y}^i(j)$ denotes the j^{th} element of $\mathbf{y}^i$. In this way, we endow each image with a vector representation in Euclidean space. Note that, all the matrices involved in this computation are sparse, and hence the computation can be performed very fast.

Once we obtain vector representations of the images, clustering is straightforward. The simplest way is to use $\mathbf{y}^1$ (called *Fiedler vector* in spectral graph theory [7]) to cut the image collection into several pieces. For details, see [19][11][14]. Another way is to use k-means clustering algorithm on the image vectors. Previous works demonstrate that spectral embedding followed by k-means can produce good result [21].

There are many applications of the embedding and clustering results. The major one should be browsing. The images can be grouped into semantic categories. Also, they can be visualized in a two-dimensional plane. Some preliminary results on small databases have been presented in [14].

6. EXPERIMENTAL EVALUATION

In this section, several illustrative examples are given. Due to the lack of sufficient resources, we are currently not able to perform image search on the whole Internet which is our ultimate goal. The purpose of this section is to provide people with an intuition on how our system works based on the techniques we described previously.

6.1 ImageSeer: The System Overview

In the above three sections, we have systematically described our techniques for web image search and organization, *i.e.* vision-based page segmentation, link and page layout based graph models and spectral analysis for image ranking, embedding and clustering. In this section, we will introduce our image search engine, *ImageSeer*.

In order to achieve fast response, the images are crawled from the web and their surrounding texts are extracted. The surrounding texts are used to index the images. We use BM2500 in Okapi as our relevance ranking function [22] which has been proved many times to be effective in information retrieval community. For details about our implementation, see [29]. Our system combines the textual relevance score and ImageRank as follows:

$$s(\mathbf{x}) = t \times \pi(\mathbf{x}) + (1-t) \times r(\mathbf{x}, \mathbf{q}) \qquad (23)$$

where $r(\mathbf{x}, \mathbf{q})$ is the textual relevance score of image $\mathbf{x}$ to query $\mathbf{q}$, $\pi(\mathbf{x})$ is the ImageRank of $\mathbf{x}$, and $s(\mathbf{x})$ is the combined score. t is a parameter, typically set to 0.1~0.2. Here, $\pi(\mathbf{x})$ and $r(\mathbf{x}, \mathbf{q})$ have been normalized into the same scale.

When the user submits a query, the system first computes the relevance score for every image and the images are ranked according to their relevance scores. For top N images, we re-rank them according to the combined scores. The re-ranked top N images are then presented to the user. Figure 4 shows the design of our system.

6.2 Comparison with PicASHOW: a simple example

Recall that PicASHOW has two basic assumptions: (a) Images which are co-contained in pages are likely to be related to the same topic. (b) Images which are contained in pages that are co-cited by a certain page are likely related to the same topic. Our system also has two assumptions listed in Section 2. Here, we give a simple example to compare PicASHOW's assumptions with our assumptions, which are the fundamental differences between our approach and PicASHOW from the perspective of link analysis.

Figure 5 shows the web page http://news.yahoo.com (left part) and three other pages (right part) it points to. The images in http://news.yahoo.com are with red frame, and we call them "red images" for the sake of simplicity. The images in other three pages are with green frame, and we call them "green images". Based on PicASHOW's assumption (a), all the red images are related to the same topics since they are contained in the same page. However, it is clear to see that they are related to different topics, *i.e.* business, entertainment, sport, technology, and politics, respectively. Also, PicASHOW's assumption (b) implies that the green images are related to the same topics since they are co-cited by http://news.yahoo.com. Again, it can be seen that the green images are related to two different topics, *i.e.* entertainment and sport.

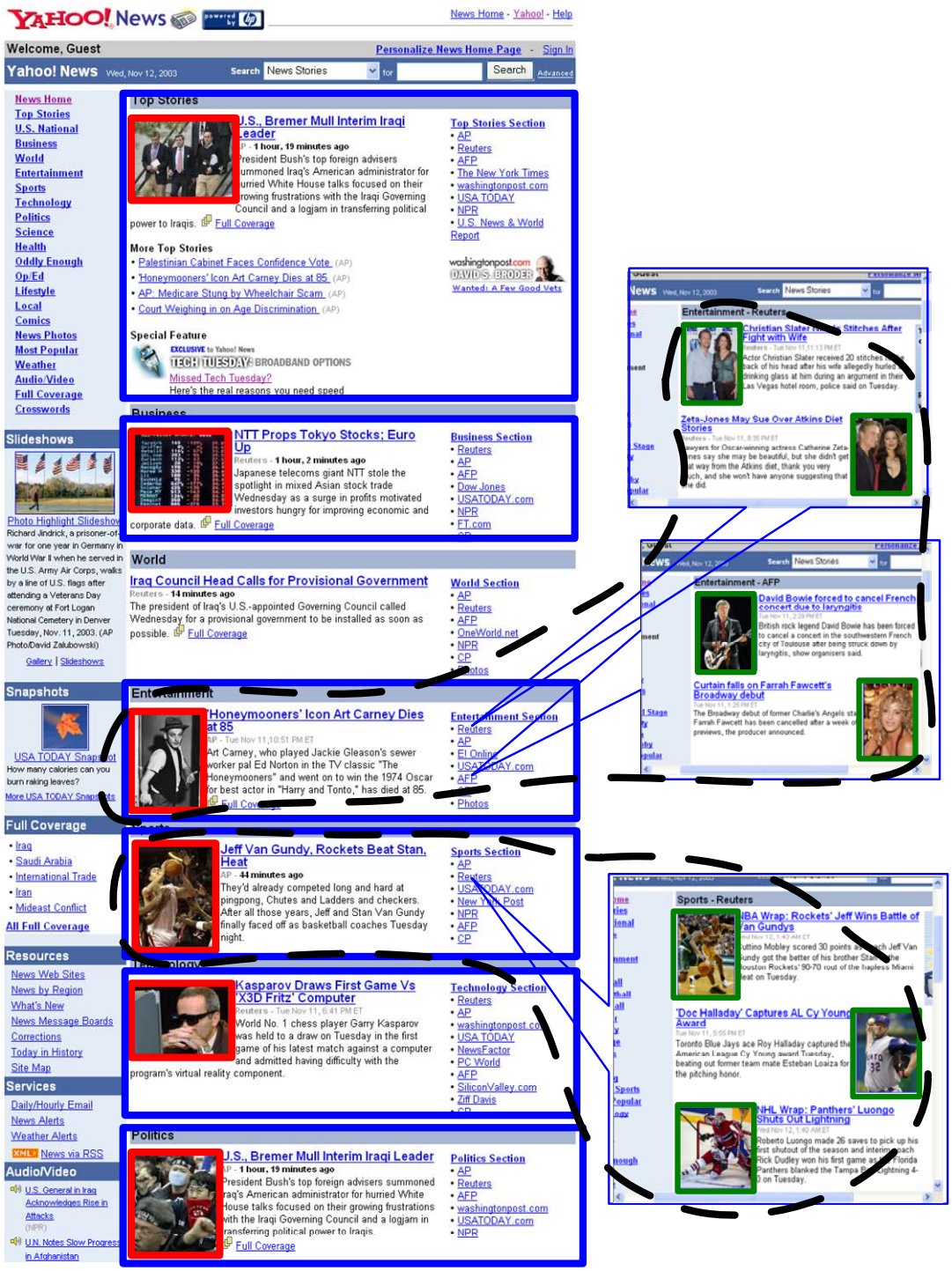

Figure 5: The web page http://news.yahoo.com (left) is segmented into blocks. The right three pages are linked by the sport block and the entertainment block. The images in the dashed circle are related to the same topic.

If we view these web pages from the block level as suggested by our approach, we get different results. First, the web page http://news.yahoo.com is segmented into semantic blocks. Thus, the red images are regarded as different semantic objects. The green images are linked by two different blocks. Thus, they are classified into two different semantic classes, as shown in the Figure 5.

This example shows that our method is possible to get more accurate information than PicASHOW.

Figure 6. Search Results of query "panda". Sorted only by relevance.

Figure 7. Search Results of query "panda". Sorted by relevance and ImageRank.

6.3 Data Preparation

All the data we used in our search experiments are crawled from the Internet. Starting from the following website

http://www.yahooligans.com/content/animals/

We crawled 32563 web pages in total by breath first crawling. From these web pages, 18678 JPG images are extracted. We filter those images whose ratio between width and height are greater than 5 or smaller than 1/5, since these kinds of images are probably of low quantity. We also remove those images whose width and height are both smaller than 45 pixels due to the same reason. Finally, we got 17134 images which form our image database. Our database mainly contains images about animal and plant.

For each web page, the VIPS page segmentation algorithm was applied to divide it into blocks. For each block, the hyperlinks were extracted. For each image, the image blocks containing that image were identified and the surrounding texts were extracted within these image blocks.

6.4 Image Search

In this subsection, we compare two methods for web image search. In the first one, we use textual information only. For each image, we compute its relevance score to the query keyword. The images are ranked according to their relevance score. In the second one, the relevance score and ImageRank are combined, and the combined score are used for image ranking. Note that, PicASHOW is based on some text search engine. It first obtained some relevant web pages, and then the images contained in these pages are ranked according to their methods. PicASHOW does not use the surrounding texts explicitly which is quite different from our approach. For these reasons, we did not compare our approach with PicASHOW on web image search in this paper.

Figure 8. Search Results of query "elephant". Sorted only by relevance.

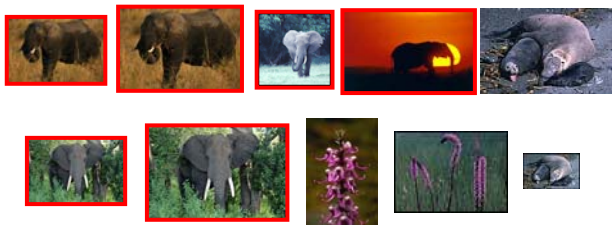

Figure 9. Search Results of query "elephant". Sorted by relevance and ImageRank.

Figure 6 and 7 shows the search results of the query "panda". For each method, the top fifteen images are returned. The images are sorted in order, from left to right and from top to bottom. In the first result obtained by using relevance score only, there are 11 images of panda (with red frame) among the top fifteen returns, as shown in Figure 6. In the second result obtained by using combined score of relevance and ImageRank, there are 14 images of panda (with red frame) among the top fifteen returns, as shown in Figure 7. As can be seen, the combined score performs better than relevance score only. This is because that the surrounding texts can not be always accurately extracted. Also, sometimes the top images have the *same* textual representations. In this case, these images are ranked at random if we only use the relevance score. The ImageRank evaluates the importance of an image on the Web. Thus, the combined score can provide a better ranking. Note that, some images have the same content but different sizes, and hence they are considered as different images. Further content-based analysis might be able to distinguish them.

The second query is "elephant". Figure 8 shows the result by using relevance score only and figure 9 shows the result by using the relevance score and ImageRank. The images are sorted in order, from left to right and from top to bottom. As can be seen, if we only use the relevance score, then there is only *one* image of elephant (with red frame) in the top 5 returns and *four* images of elephant (with red frame) in the top 10 returns. The second, forth, fifth, sixth and seventh images are actually "elephant tree" (kind of plant) and the third image is "elephant seal". When ImageRanks are applied, these two numbers increase to *four* and *six*, respectively.

We are currently conducting performance evaluation by having a large number of users provide their ranking on the retrieved result. We hope to incorporate this result into our paper if it is accepted by the conference.

6.5 Discussions

Our experimental results showed that image search results can be improved by using ImageRank. However, it is important to note that this is not always the case. In fact, for many cases, the improvement by using ImageRank is minor. This might be due to the following facts,

1. Our image database is still small compared to the Web and hence the obtained link information is spare and incomplete.
2. Due to the consideration of computational complexity, a single image graph is constructed and the ImageRanks are induced from this graph. However, the image database always contains many semantic classes. For some semantic class, there might be only few images with few hyperlinks. Thus, the ImageRank values computed for these images will be inevitably small yet there is no convincing evidence that these images are less important. When computational complexity is not a concern, we can first cluster the images into semantic classes and the ImageRank is then computed within each class.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we described a WWW image search engine called *ImageSeer* and focused on three fundamental problems, i.e. representation, similarity measure and ranking. For representation, two schemes were proposed. One is based on the textual representation obtained by surrounding text and image file title, etc. The other is based on a vector representation obtained from the image graph model such that if two images have strong link relationship then they are close to each other in the vector space. By constructing the image graph, the weights on the edge give the similarity measures between the two images. We do not consider the visual similarity measure in this paper because it is computationally extensive. By using the notion of random walk on the graph, we compute ImageRanks which is combined with the textual relevance scores for ranking.

Several questions remain to be investigated in our future work.

1. In this paper, the relationships between web pages, blocks and images are interpreted as *set* structure rather than *tree* structure. However, it might be more natural to interpret a web page as a tree and the images can be viewed as the leaf nodes of the tree. It remains unclear how to incorporate the tree structure into our graph models.
2. Several graph models are constructed exclusively from the link structure and page layout. The textual information can also be incorporated into the graph models if computational complexity is not a concern.
3. Within our framework of analysis, the page-to-page graph is induced as a byproduct. It is used to induce the block-to-block graph and image-to-image graph. In fact, we can also compute PageRank from this graph. Some theoretical analysis shows that this PageRank coincides with the original PageRank [4] if and only if the block importance function (see Section 4.2) is defined as the number of links contained in the block.

8. REFERENCES

[1] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering", *Advances in Neural Information Processing Systems* 14, Canada, 2001.

[2] C. M. Bishop, Neural networks for pattern recognition, Clarendon Press, 1995.

[3] C. Brew and S. Schulte im Wade, "Spectral clustering for german verbs", *Proc. of the Conf. on Empirical Methods in Natural Language Processing*, Philadelphia, PA, 2002,

[4] S. Brin and L. Page, "The anatomy of a large-scale hypertextual (Web) search engine", In *The Seventh International World Wide Web Conference*, 1998.

[5] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma, "Extracting content structure for web pages based on visual representation", *Proc.5th Asia Pacific Web Conference*, Xi'an China, 2003.

[6] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma, "VIPS: a vision-based page segmentation algorithm", Microsoft Technical Report, MSR-TR-2003-79, 2003.

[7] Fan Chung. Spectral graph theory. *Regional conference series in mathematics*, no 92, American Mathematical Society, Providence, RI, 1997.

[8] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. harshman, "Indexing by latent semantic analysis", *Journal of the American Society of Information Science*, 41(6):391-407, 1990.

[9] G. Flake, S. Lawrence, L. Giles, and F. Coetzee, "Self-organization and identification of web communities", *IEEE Computer*, pp. 66-71, 2002.

[10] C. Frankel, M. Swain, and V. Athitsos, "WebSeer: An image search engine for the world wide web", *TR-96-14*, Department of Computer Science, University of Chicago, 1996.

[11] S. Guattery and G. L. Miller, "Graph embedding and laplacian eigenvalues", *SIAM Journal on Matrix Analysis and Applications*, 21(3)-2000.

[12] X. He, W.-Y. Ma, K. Oliver, M. Li, and H.-J. Zhang, "Learning and inferring a semantic space from user's relevance feedback for image retrieval", *Proc. 10th ACM Conference on Multimedia*, 2002.

[13] X. He, K. Oliver, W.-Y. Ma, M. Li, and H.-J. Zhang, "Learning a semantic space from user's relevance feedback for image retrieval", *IEEE Trans. on Circuit and Systems for Video Technology*, Vol 13, No. 1, Jan 2003.

[14] X. He, W.-Y. Ma, and H. J. Zhang, "ImageRank: spectral techniques for structural analysis of image database", *IEEE International Conference on Multimedia and Expo*, Baltimore, Maryland, 2003.

[15] J. Kleinberg, "Authoritative sources in a hyperlinked environment", *Proc. 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[16] R. Lempel and A. Soffer, "PicASHOW: Pictorial authority search by hyperlinks on the web", *Proc. 10th Int. World Wide Web Conf.*, pp. 438-448, Hong Kong, China, 2001.

[17] J. Li, J. Z. Wang, and G. Wiederhold, "IRM: integrated region matching for image retrieval", *Proc. ACM Conf. Multimedia*, Los Angeles, 2000.

[18] W.-Y. Ma and B. S. Manjunath, "Texture features and learning similarity", *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 425-430, San Francisco, CA, !996.

[19] B. Mohar, "Some applications of Laplace eigenvalues of graphs", In *Graph Symmetry: Algebraic Methods and Applications* (G. Hahn, G. Sabidussi, eds.), volume 497 of NATO ASI Series C, pages 227-275, Kluwer, Dordrecht, 1997.

[20] W.-Y. Ma and B. S. Manjunath, "Netra: a toolbox for navigating large image databases", *Multimedia Systems*, vol. 7, no. 3, pp. 184-189, 1999.

[21] Andrew Y. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm", *Advances in Neural Information Processing Systems* 14, Vancouver, Canada, 2001.

[22] S. E. Robertson and S. Walker, "Okapi/Keenbow at TREC-8", in *the eighth Text Retrieval Conference* (TREC 8), 1999, pp. 151-162.

[23] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: a power tool for interactive content based image retrieval", *IEEE Trans. Circuit and Systems for Video Technology*, 8(5), pp. 644-655, 1998.

[24] S. Sclaroff, L. Taycher, and M. La Cascia, "ImageRover: a content-based image browser for the world wide web", in *IEEE workshop on content-based access of image and video libraries*, pages 2-9, San Juan, Puerto Rico, June 1994.

[25] J. Shi and J. Malik, "Normalized cuts and image segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), pp. 888-905, 2000.

[26] J. Smith and S.-F. Chang, "VisualSEEk: a fully automated content-based image query system", *Proc. ACM Conf. Multimedia*, New York, 1996.

[27] M. Stricker and M. Orengo, "Similarity of color images", *Proc. Storage and Retrieval for Image and Video Databases*,*SPIE 2420*, pp. 381-392, 1995.

[28] V. Vapnik, The nature of statistical learning theory, Springer, New York, 1995.

[29] J. R. Wen, R. Song, D. Cai, K. Zhu, S. Yu, S. Ye, and W.-Y. Ma, "Microsoft Research Asia at the web track of TREC 2003", in *the twelfth Text Retrieval Conference* (TREC 12), 2003.

[30] S. Yu, D. Cai, J.-R. Wen, and W.-Y. Ma, "Improving pseudo-relevance feedback in web information retrieval using web page segmentation", *Proc. 12th World Wide Web Conference*, Budapest, Hungary, 2003.

[31] C. Lee, W.-Y. Ma and H.-J. Zhang, "Information embedding based on user's relevance feedback for image retrieval", *Technical Report*, HP Labs, 1998.

[32] M. Li, Z. Chen, W. Liu and H.-J. Zhang, "A statistical correlation model for image retrieval", *in Proc. 3rd International Workshop on Multimedia Information Retrieval*, Ottawa, Canada, 2001, pp. 42-45, 2001.

[33] J. Smith and S.-F. Chang, "WebSEEK, a content-based image and video search and catalog tool for the web", *IEEE Multimedia*, 1997.

[34] Andrew Y. Ng, A. X. Zheng and M. Jordan, "Stable algorithms for link analysis", in *Proc. 24th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2001.

[35] X. He, S. Yan, Y. Hu and H. J. Zhang, "Learning a locality preserving subspace for visual recognition", in *Proc. IEEE International Conference on Computer Vision*, Nice, France, 2003.

[36] G. Jeh and J. Widom, "Scaling personalized web search", in *Proc. of the twelfth International World Wide Web Conference*, 2003.

[37] T. H. Haveliwala, "Topic-sensitive PageRank", in *Proc. of the eleventh International World Wide Web Conference*, Hawaii, USA, 2002.