Searching for Logo and Trademark Images on the Web

Euripides G.M. Petrakis† Epimenides Voutsakis† Evangelos E. Milios‡
petrakis@intelligence.tuc.gr pimenas@softnet.tuc.gr eem@cs.dal.ca

† Dept. of Electronic and Comp. Engineering, Technical University of Crete (TUC), Chania, Greece
‡ Faculty of Comp. Science, Dalhousie University, Halifax, Nova Scotia, Canada

ABSTRACT

This work shows that it is possible to exploit text and image content characteristics of logo and trademark images in Web pages for enhancing the performance of retrievals on the Web. Searching for important (authoritative) Web pages and images is a desirable feature of many Web search engines and is also taken into account. State-of-the-art methods for assigning higher ranking to important Web pages, over other Web pages satisfying the query selection criteria, are considered and evaluated. PicASHOW exploits this idea in retrieval of important images on the Web using link information alone. WPicASHOW (Weighted PicASHOW), is a weighted scheme for co-citation analysis incorporating within the link analysis method of PicASHOW the text and image content of the queries and of the Web pages. The experimental results demonstrate that Web search methods utilizing content information (or combination of content and link information) perform significantly better than methods using link information alone.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Abstracting methods, Indexing methods*; I.5.4 [**Pattern Recognition**]: Applications—*Computer Vision*

General Terms

Performance, Experimentation, Algorithms

Keywords

image retrieval, logo, trademark, feature extraction, authority, link analysis

1. INTRODUCTION

The World Wide Web is host to millions of images on every conceivable topic. The images are used to enhance the information content of Web pages, capture the attention of users or to reduce the textual content of Web sites. In scientific, artistic, technical, or corporate Web sites, images comprise the majority of digital content and are characteristic of the content and type of these Web sites.

Searching for effective methods to retrieve images from the Web has been in the center of many scientific efforts during the last few years [9]. The relevant technology evolved rapidly also thanks to prior advances in Web systems technology [1], information retrieval [16] and image database research [20, 17]. Several approaches to the problem of content-based image retrieval on the Web have been proposed and some have been implemented on research prototypes (e.g., PicToSeek [6], ImageRover [24],WebSEEK [21], Diogenis [2]) and commercial systems. The last category of systems, includes general purpose image search engines such as Google Image Search [1], Yahoo [2], Altavista [3], Ditto [4] etc.) as well as systems providing specific services to users such as unauthorized use of images (e.g., CreativePro [5]), Web and e-mail content filters, systems for image authentication (e.g., Dicontas [6]), licensing and advertising (e.g., Corbis [7]).

This work deals with the problem of retrieval of logo and trademark images on the Web. Logos and trademarks are important characteristic signs of corporate Web sites or of products presented there. A recent analysis of Web content [7] reports that logos and trademarks comprise 32,6% of the total number of images on the Web. Therefore, retrieval of logo and trademarks is of significant commercial interest (e.g., Patent Offices provide services on unauthorized uses of logos and trademarks).

The contribution of this work is not only in using existing technology for solving the retrieval problem but also, in showing how to exploit the content characteristics of logo and trademarks for enhancing the performance of retrievals on the Web. Retrieval by image content, in particular, requires integration of text and image based approaches for analyzing the content of Web pages.

Logo and trademark images are easier (than natural images) to describe by low level features (intensity, frequency histograms and features computed on the above types of histograms). Because images on the Web are not properly categorized, filters based on machine learning by decision trees for distinguishing logo and trademark images from images

CIVR'07, July 9–11, 2007, Amsterdam, The Netherlands.

[1] http://www.google.com/imghp
[2] http://images.search.yahoo.com
[3] http://www.altavista.com/image
[4] http://www.ditto.com
[5] http://www.creativepro.com
[6] http://www.dicontas.co.uk
[7] http://www.corbis.com

of other categories (e.g., graphics, photographs, diagrams, landscapes) are designed and implemented. The decision tree demonstrated classification accuracy as high as 85%.

Once logo and trademark images are detected, effective content-based image retrieval on the Web often requires that important (authoritative) images satisfying the query selection criteria are assigned higher ranking over other relevant images. This is achieved by exploiting the results of link analysis for re-ranking the results of retrieval. Classical link analysis methods such as *HITS* [10] and *PageRank* [13] estimate the quality of Web pages and the topic relevance between the Web pages and the query. These methods estimate the importance of Web pages as a whole. PicASHOW [11], in particular, shows how to estimate the importance of images contained within Web pages. However, PicASHOW does not show how to handle image content and queries by image example. This is solved by WPicASHOW [26] (Weighted PicASHOW) a weighted scheme for co-citation analysis that incorporates, within the link analysis method of PicASHOW, the text and image content of the queries and of the Web pages.

The methods referred to above are implemented and evaluated in *IntelliSearch* [8] [25], a complete and fully automated information retrieval system for the Web. It supports fast and accurate responses to queries addressing text and images in Web pages by incorporating state-of-the-art image indexing and retrieval methods by text (e.g., the Vector Space Model) in conjunction with efficient ranking of Web pages and images by importance (authority) such as WPicASHOW. *IntelliSearch* stores a crawl of the Web with more than 1,5 million Web pages with images. It offers an ideal test-bed for experimentation and training and serves as a framework for a realistic evaluation of many Web image retrieval methods. The experimental results demonstrate that giving higher ranking to important images seems to reduce the accuracy of retrievals (the important images are not always the most relevant ones).

Existing approaches for handling logos and trademarks [8, 12] focus entirely on image content analysis and high precision answers to queries by image example on stand-alone data sets. They don't focus on detection (i.e., discrimination between trademark and not trademark images) nor do they show how to retrieve high quality answers from the Web.

The rest of this paper is organized as follows: Extraction of meaningful image descriptions from Web pages and image similarity measures based on the matching of image descriptions are discussed in Section 2. PicASHOW and WPicASHOW, the image authority searching methods considered in this work are presented in Section 3. *IntelliSearch*, a content-based retrieval of logo and trademark images that integrates the above ideas is presented in Section 4. Experimental results are presented and discussed in Section 5 followed by conclusions in Section 6.

2. IMAGE CONTENT REPRESENTATION

Logo and trademark images are easier to describe by low level features (e.g., color histograms, text features). The focus of this work is not on novel image feature extraction but on showing how to search for logo and trademarks on the Web for a given and well established set of features (such as those used in [8, 12]).

[8]http://www.intelligence.tuc.gr/intellisearch

2.1 Text Features

Typically, images are described by the text surrounding them in the Web pages [19]. The following types of image descriptive text are derived based on the analysis of `html` formatting instructions:

Image Filename: The `URL` entry (with leading directory names removed) in the `src` field of the `img` formatting instruction.

Alternate Text: The text entry of the `alt` field in the `img` formatting instruction. This text is displayed on the browser (in place of the image), if the image fails to load. This attribute is optional (i.e., is not always present).

Page Title: The title of the Web page in which the image is displayed. It is contained between the `TITLE` formatting instructions in the beginning of the document. It is optional.

Image Caption: A sentence that describes the image. It usually follows or precedes the image when it is displayed on the browser. Because it does not correspond to any `html` formatting instruction it is derived either as the text within the same table cell as the image (i.e., between `td` formatting instructions) or within the same paragraph as the image (i.e., between `p` formatting instructions). In either case, the caption is limited to 30 words before or after the reference to the image file. If neither case applies, the caption is considered to be empty.

The following are two examples of `html` code, both with a reference to image *"logo.gif"*.

- *. . . </td> <td> Our company's logo <img src="logo.gif" alt="software logo">
 . . . </td> <td> is registered since 1990 . . . </td> <td>.*
- *. . . Our company's logo logo is registered since 1990 .*

All descriptions are lexically analyzed and reduced into term (noun) vectors. First, all terms are reduced into their morphological roots, using the Porter [15] suffix stripping (stemming) algorithm. Similarly, text queries are also transformed to term vectors and matched against image term vectors according to the vector space model. More specifically, the similarity between the query Q and the image I is computed as a weighted sum of similarities between their corresponding term vectors

$$\begin{aligned} &S_{text}(Q,T) = \\ &S_{file\_name}(Q,I) \quad + \quad S_{alternate\_text}(Q,I) \quad + \\ &S_{page\_title}(Q,I) \quad + \quad S_{image\_caption}(Q,I). \end{aligned} \tag{1}$$

Each S term is computed as a weighted sum of $tf \cdot idf$ terms without normalizing by query term frequencies (it is not required for short queries). All measures above are normalized in [0,1].

2.2 Image Features

Image content is described in terms of features computed from raw images. All images are converted to grey scale. For logo and trademark images the following features are computed:

Intensity Histogram: Shows the distribution of intensities over the whole range of intensity values ([0..255] in this work).

Energy Spectrum [22]: Describes the image by its frequency content. It is computed as a histogram showing the distribution of average energy over 256 co-centric rings (with the largest ring fitting the largest inscribed circle of the DFT spectrum).

Moment Invariants [23]: Describes the image by its spatial arrangement of intensities. It is a vector of 7 moment coefficients.

The above representations are used to solve the following two problems:

Logo-Trademark Detection: A five-dimensional vector is formed from each image: Each image is specified by the mean and variance of its Intensity and Energy spectrums plus a count of the number of distinct intensities per image. A set of 1,000 image examples is formed consisting of 500 logo-trademark images and 500 images of other types. Images of other types can belong to more than one class: non-logo graphics, photographs, diagrams etc. Their feature vectors are fed into a decision-tree [27] which is trained to detect logo and trademark images. The estimated classification accuracy by the algorithm is 85%. For each image the decision computes an estimate of its likelihood of being logo or trademark or "Logo-Trademark Probability".

Logo-Trademark Similarity: The similarity between two images Q, I (e.g., query and a Web image) is computed as

$$S_{image}(Q,I) = S_{intensity\_spectrum}(Q,I) + S_{energy\_spectrum}(Q,I) + S_{moment\_invariants}(Q,I). \quad (2)$$

The similarity between histograms is computed by their intersection [5] whereas the similarity between their moment invariant is computed as *1 - Euclidean_vector_distance*.

All measures above are normalized to lie in the interval [0, 1]. To answer queries consisting of both text and example image, the similarity between a query Q and an image I is computed as

$$S = \lambda S_{image}(Q,I) + (1-\lambda) S_{text}(Q,I), \quad (3)$$

where λ denotes the relative significance of image and text descriptions. In this work $\lambda = 0.5$. More appropriate weights may be specified by machine learning.

3. IMAGE LINK ANALYSIS METHODS

Co-citation analysis is proposed as a tool for assigning importance to pages or for estimating the similarity between a query and a Web page. The main idea behind this approach is that a link from page a to page b may be regarded as a reference from the author of a to b.

The number and quality of references to a page provide an estimate of the quality of the page and also a suggestion of relevance of its contents with the contents of the pages pointing to it. *HITS* [10] exploits this information to estimate the relevance between a query and a Web page and

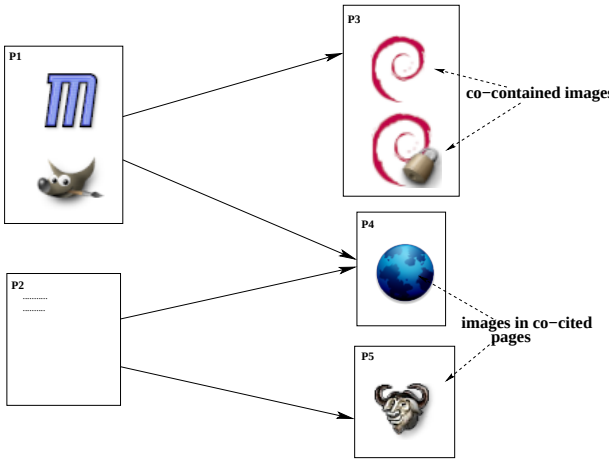

Figure 1: An example of a focused graph with co-contained and co-cited images.

ranking of this page among other relevant pages. Building upon the same idea, PicASHOW [11] demonstrates how to retrieve high quality Web images on the topic of a keyword-based query. It does not show how to process queries by example image. This is exactly the focus of this work.

PicASHOW relies on the idea that images co-contained or co-cited by Web pages are likely to be related to the same topic. Figure 1 illustrates examples of co-contained and co-cited images. *PicAHOW* computes authority and hub values by link analysis on the *query focused graph* $\mathcal{F}$ (i.e., a set of pages formed by initial query results expanded by backward and forward links). Non-informative images such as banners and "stop images" (e.g., bars, buttons, mailboxes) are filtered out from $\mathcal{F}$ utilizing simple heuristics (e.g., small file size).

PicASHOW introduces the following adjacency matrices defined on the set of pages in the query focused graph:

$\mathcal{W}$: The page to page adjacency matrix (as in HITS) relating each page in $\mathcal{F}$ with the pages it points to. The rows and the columns in $\mathcal{W}$ are indices to pages in $\mathcal{F}$. Then, $w_{ij} = 1$ if page i points to page j; 0 otherwise.

$\mathcal{M}$: The page to image adjacency matrix relating each page in $\mathcal{F}$ with the images it contains. The rows and the columns in $\mathcal{M}$ are indices to pages and images in $\mathcal{F}$ respectively. Then, $m_{ij} = 1$ if page i points to (or contains) image j.

$(\mathcal{W}+\mathcal{I})\mathcal{M}$: The page to image adjacency matrix ($\mathcal{I}$ is the identity matrix) relating each page in $\mathcal{F}$ both, with the images it contains and with the images contained in pages it points to.

Similarly to HITS, PicASHOW defines the so called image *co-citation* $[(\mathcal{W}+\mathcal{I})\mathcal{M}]^T \cdot (\mathcal{M}+\mathcal{I})\mathcal{W}$ and *bibliographic* $(\mathcal{W}+\mathcal{I})\mathcal{M}) \cdot [(\mathcal{W}+\mathcal{I})\mathcal{M}]^T$ matrices respectively. The ij-th entry of the image co-citation matrix is the number of pages that jointly point to images with indices i and j. The ij-th entry of the image bibliographic matric is the number of images jointly referred to by pages i and j. PicASHOW computes the answers to a query by ranking the elements of the principal eigenvector of the image co-citation matrix by their authority values.

	P_1	P_2	P_3	P_4	P_5
P_1	0	0	1	1	0
P_2	0	0	0	1	1
P_3	0	0	0	0	0
P_4	0	0	0	0	0
P_5	0	0	0	0	0

						
P_1	0	0	1	1	0	0
P_2	0	0	0	0	0	0
P_3	1	1	0	0	0	0
P_4	0	0	0	0	1	0
P_5	0	0	0	0	0	1

						
P_1	1	1	1	1	1	0
P_2	0	0	0	0	1	1
P_3	1	1	0	0	0	0
P_4	0	0	0	0	1	0
P_5	0	0	0	0	0	1

Figure 2: Adjacency matrices $\mathcal{W}$, $\mathcal{M}$ and $(\mathcal{W}+\mathcal{I})\mathcal{M}$ computed by PicASHOW for the focused graph of Figure 1.

Image						
Authorities	0.492	0.492	0.339	0.339	0.519	0.117

Page	P_1	P_2	P_3	P_4	P_5
Hubs	0.519	0.0001	0.854	0.001	0

Figure 3: Image Authority (top) and Hub values (bottom) computed by PicASHOW in response to query "Debian trademark".

Figure 2 illustrates these matrices for the pages $(P_1, P_2, \ldots P_5)$ and images of Figure 1. Notice that, in PicASHOW all non-zero values in $\mathcal{M}$, $\mathcal{W}$ and $(\mathcal{W}+\mathcal{I})\mathcal{M}$ matrices are 1 (non normalized weights). Figure 3 illustrates authority and hub values computed by PicASHOW in response to query "Debian logo". Notice the high authority scores of pages showing logo or trademark images of "Debian Linux". Notice that Mozilla trademark has higher authority value than Debian trademark.

Hub and Authority values of images are computed as the principal eigenvectors of the image-co-citation $[(\mathcal{W}+\mathcal{I})\mathcal{M}]^T \cdot (\mathcal{W}+\mathcal{I})\mathcal{M}$ and bibliographic matrices $(\mathcal{W}+\mathcal{I})\mathcal{M}) \cdot [(\mathcal{W}+\mathcal{I})\mathcal{M}]^T$ respectively. The higher the authority value of an image the higher its likelihood of being relevant to the query.

PicASHOW can answer queries on a given topic but, similarly to $HITS$, it suffers from the following problems [4]:

Mutual reinforcement between hosts: Encountered when a single page on a host points to multiple pages on another host or the reverse (when multiple pages on a host point to a single page on another host).

Topic drift: Encountered when the query focused graph contains pages not relevant to the query. Then, the highest authority and hub pages tend not to be related to the topic of the query.

PicASHOW does not handle mutual reinforcement between nodes (except that it constraints the number of references per image to one by identifying replicated images) and topic drift nor does it handle queries by example. WPicASHOW handles all these issues:

Mutual reinforcement is handled by normalizing the weights of nodes pointing to k by $1/k$. Similarly, the weights of all l pages pointing to the same page are normalized by $1/l$. An additional improvement is to purge all intra-domain links except links from pages to their contained images.

Topic Drift is handled by regulating the influence of nodes by setting weights on links between pages. The links of the page-to-page relation $\mathcal{W}$ are assigned a relevance value computed according to the vector space model as the similarity between the term vector of the query and the term vector of the anchor text on the link between the two pages. The weights of the page-to-image relation matrix $\mathcal{M}$ are computed depending on query type: For text (e.g., keyword) queries the weights are computed according to Equation 1 (as the similarity between the query and the descriptive text of an image). For queries combining text and image example, the weights are computed according to Equation 3 (as the average of similarities between the text and image contents of the query and the image respectively).

	P_1	P_2	P_3	P_4	P_5
P_1	0	0	.6	.1	0
P_2	0	0	0	.1	.1
P_3	0	0	0	0	0
P_4	0	0	0	0	0
P_5	0	0	0	0	0

						
P_1	0	0	.1	.1	0	0
P_2	0	0	0	0	0	0
P_3	.8	.7	0	0	0	0
P_4	0	0	0	0	.2	0
P_5	0	0	0	0	0	.15

						
P_1	.48	.42	.1	.1	.02	0
P_2	0	0	0	0	.02	.015
P_3	.8	.7	0	0	0	0
P_4	0	0	0	0	.2	0
P_5	0	0	0	0	0	.15

Figure 4: Adjacency matrices $\mathcal{M}$, $\mathcal{W}$ and $(\mathcal{M}+\mathcal{I})\mathcal{W}$ computed by WPicASHOW for the focused graph of Figure 1.

Queries may be formulated either by keywords (or phrases) or by a combination of keywords and image ex-

Image						
Authorities	.751	.657	.0418	.0418	.008	0

Page	P_1	P_2	P_3	P_4	P_5
Hubs	.519	.0001	.854	.001	0

Figure 5: Image Authority (top) and Hub values (bottom) computed by WPicASHOW in response to query "Debian logo".

ample. In both cases of image queries, WPicASHOW starts by formulating the query focused graph as follows:

- An initial set S of images is retrieved. These are images contained or pointed to by pages matching the query keywords according to Equation 1.
- Stop images (banners, buttons, etc.) and images with logo-trademark probability less than 0.5 are ignored. At most T images are retained and this limits the size of the query focused graph ($T = 10,000$ in this work).
- The set S is expanded to include pages pointing to images in S.
- The set S is further expanded to include pages and images pointed to by pages already in S. To limit the influence of very popular sites, for each page in S, at most t ($t = 100$ in this work) new pages are included.
- The last two steps are repeated until S contains T pages and images.

WPicASHOW then builds $\mathcal{M}$, $\mathcal{W}$ and $(\mathcal{W} + \mathcal{I})\mathcal{M}$ matrices for information in S. Figure 4 illustrates these matrices for the same pages and images of Figure 1 with weights corresponding to query "Debian logo".

Figure 3 illustrates authority and hub values computed by WPicASHOW in response to query "Debian logo". The trademark images of "Debian Linux" are assigned the highest authority values followed by the images of Mozilla Firefox.

4. INTELLISEARCH

IntelliSearch [25] is implemented in Java under Linux. Figure 6 illustrates the architecture of the proposed system. *IntelliSearch* consists of several modules, the most important of them being the following:

Crawler module: Implemented based upon Larbin [9], the crawler assembled locally a collection of 1,5 million pages with images. The crawler started its recursive visit of the Web from a set of 14,000 pages which is assembled from the answers of Google image search[10] to 20 queries on topics related to Linux and Linux products. The crawler worked recursively in breadth-first order and visited pages up to depth 5 links from each origin.

[9]http://larbin.sourceforge.net
[10]http://www.google.com/imghp

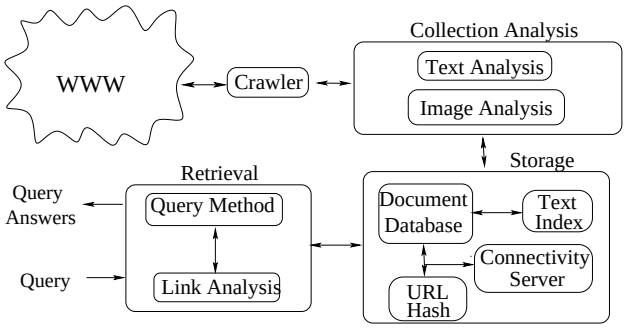

Figure 6: *IntelliSearch* Architecture.

Collection analysis module: The content of crawled pages is analyzed. Text, images, link information (forward links) and information for pages that belong to the same site is extracted.

Storage module: Implements storage structures and indices providing fast access to Web pages and information extracted from Web pages (i.e., text, image descriptions and links). For each page, except from raw text and images, the following information is stored and indexed: Page URLs, image descriptive text (i.e., alternate text, caption, title, image file name), terms extracted from pages, term inter document frequencies (i.e., term frequencies in the whole collection), term intra document frequencies (i.e., term frequencies in image descriptive text parts), link structure information (i.e., backward and forward links). Image descriptions are also stored.

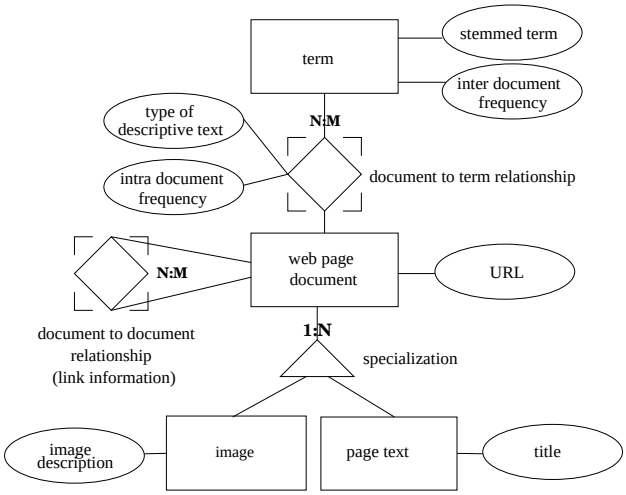

Figure 7: The Entity Relational Diagram (ERD) of the database.

The Entity Relationship Diagram (ERD) of the database in Fig. 7 describes entities (i.e., Web pages) and relationships between entities. There are many-to-many (denoted as $N : M$) relationships between Web pages implied by the Web link structure (by forward and backward links), one-to-many (denoted as

$1:N$) relationships between Web pages and their constituent text and images and $N:M$ relationships between terms in image descriptive text parts and documents. The ERD also illustrates properties of entities and relationships (i.e., page URLs for documents, titles for page text, image content descriptions for images, stemmed terms, inter and intra document frequencies for terms in image descriptive text parts).

The database schema is implemented in BerkeleyDB [11] Java Edition. BerkeleyDB is an embedded database engine providing a simple Application Programming Interface (API) supporting efficient storage and retrieval of Java objects. The mapping of the ERD of Fig. 7 to database files (Java objects) was implemented using the Java Collections-style interface. Apache Lucene[12] is providing mechanisms (i.e., inverted files) for indexing text and link information. There are Hash tables for URLs and inverted files for terms and link information. Two inverted files implement the connectivity server [3] and provide fast access to linkage information between pages (backward and forward links) and two inverted files associate terms with their intra and inter document frequencies and allow for fast computation of term vectors.

Retrieval module: Image queries on the Web are issued through the user interface by specifying keywords (or free text) or by specifying a combination of example image and text. As it is typical in Web search engines, text queries addressing text Web pages are also supported. In the case of text queries the system returns images in Web pages on the topic of the query (e.g., typically images associated with similar keywords or text as descriptions). The highest complexity of image queries is encountered in the case of queries by image example. In this case, the system returns similar images on the topic of the query. The user is prompted at the user interface to select mode of operation (query type, retrieval of text pages or image retrieval). All methods in Sec. 2.1 are implemented and can be used to search the database.

5. EXPERIMENTS

Different image retrieval methods are implemented and evaluated. The competitor methods are:

PicASHOW [11]: Ranks Web images by exploiting cocitation information only. It can answer only text queries. Queries by example image (image queries) are not supported.

WPicASHOW (weighted PicASHOW) [26]: Extends PicASHOW to take into account, in addition to link information, the text and image content of both the queries and of the Web pages. It can answer both text and image queries.

Vector Space Model (VSM) [18]: Text queries are transformed to term vectors and matched against term vectors extracted from database images. The similarity between a query and a database image is computed according to Equation 1. To answer queries specifying text and image example, the similarity between a query and a database image is computed according to Equation 3.

For the evaluations, 20 characteristic queries of each type are created on topics related to Linux and Linux products. For each query the top 30 answers are retrieved. All performance results are averages over 20 queries. The evaluation is based on human relevance judgments by a human subject. For each method, the subject inspected the answers of each query and, for each answer, judged if it is similar to the query or not. This is a highly subjective process. Two or more methods may retrieve the same answer for the same query, but the same answer (by mistake) may not be recognized as similar when it is retrieved by different methods. To obtain consistent evaluations a query and a retrieved image are considered as similar if they are taken as similar by at least one method.

To evaluate the effectiveness of each candidate method, the following quantities are computed:

Precision, the percentage of relevant images retrieved with respect to the number of retrieved images.

Recall, the percentage of relevant images retrieved with respect to the total number of relevant images in the database. Due to the large size of the data set, it is practically impossible to compare every query with each database image. To compute recall, for each query, the answers obtained by all candidate methods are merged and this set is considered to contain the total number of correct answers [14].

In the following, a *precision-recall plot* is presented for each experiment. The horizontal axis in such a plot corresponds to the measured recall while the vertical axis corresponds to precision. Each method is represented by a curve. Each query retrieves the best 30 answers (best matches) and each point in a curve is the average over 20 queries. Precision and recall values are computed after each answer (from 1 to 30) and therefore, each curve contains exactly 30 points. The top-left point of a precision/recall curve corresponds to the precision/recall values for the best answer or best match (which has rank 1) while the bottom right point corresponds to the precision/recall values for the entire answer set.

A method is better than another if it achieves better precision and recall. It is possible for two (or more) precision-recall curves to intersect. This means that one of the methods performs better for small answer sets (containing less answers than the answer set at the intersection) while the other performs better for larger answer sets. The method achieving higher precision and recall for large answer sets is considered to be the best method (the typical users retrieve more than 10 or 20 images on the average).

5.1 Text Queries

In this experiment, each query is specified by a set of keywords. All queries specified the term "logo". An image in the answer is considered similar to the query if they are at the same topic (e.g., query "Linux logo" may retrieve the logo of any Linux distribution (e.g., "Debian Linux").

Figure 8 illustrates the precision/recall diagram of the three candidate retrieval methods for text queries. PicASHOW is obviously the worst method. This result indicates that link information alone is not an effective descriptor for

[11] http://www.sleepycat.com
[12] http://lucene.apache.org

image content. The answers indeed contain a lot of irrelevant images. These are images that coexist within the same high quality pages with other relevant images, or are pointed to by high quality pages (e.g., pages of software companies). WPicASHOW is more effective than PicASHOW achieving up to 20% better recall and 15% better precision. WPicASHOW assigned higher ranking to images whose surrounding text is more relevant to the topic of the query. However, VSM is the most effective method (except from the first 3 answers). This result indicates that the surrounding text is a very effective descriptor of the image itself. This method assigned higher ranking even to images contained or pointed to by very low quality pages such as pages created by individuals or small companies.

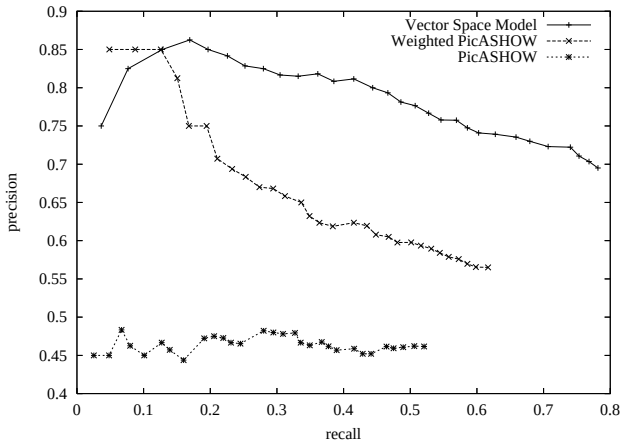

Figure 8: Precision-recall diagram for text queries corresponding to PicASHOW, WPicASHOW and the Vector Space Model.

5.2 Image Queries

Each query specifies a set of keywords along with an example logo image. For each keyword query, an appropriate logo is used as query. An image in the answer is considered similar only if its similar with the query image (e.q., query "Linux logo" with the penguin logo may only retrieve images showing a Linux penguin logo.

PicASHOW cannot answer such queries. Figure 9 illustrates the precision/recall diagram of the remaining two methods. The Vector Space Model is obviously far more effective than WPicASHOW. An important observation is that the performance gap between the two methods is wider than that of keyword queries. A closer look into the answers reveals that link analysis assigned higher ranking to Web pages with more general content on the topic of the query. The reason for this behavior is that Web pages with more general content are more strongly connected than pages with more specific topic. In this experiment, with the addition of logo image, the queries become more specific than before and WPicASHOW assigned higher ranking to more general but irrelevant images although in many cases these images are somehow related to the topic of the query (e.g., the "GNU" head logo with the "FSF" logo).

This behavior is in fact common to any link analysis method. WPicASHOW, as any other link analysis method, assigned higher ranking to higher quality but not necessary relevant pages. High quality pages, on the other hand, may be irrelevant to the content of the query. WPicASHOW attempted to compromise between the two.

The size of the data set is also a problem in both experiments. If the queries are very specific, the set of relevant answers is small and within it, the set of high quality and relevant answers are even smaller. The results may improve with the size of the data set, implying that it is plausible for the method to perform better when applied to the whole Web.

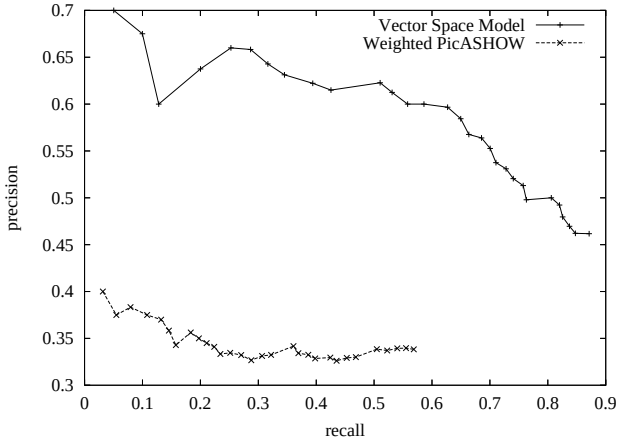

Figure 9: Precision-recall diagram for image queries corresponding to WPicASHOW and the Vector Space Model.

6. CONCLUSIONS AND FUTURE WORK

Existing approaches for handling logos and trademarks (e.g., [8, 12]) focus entirely on image content analysis and high precision answers to queries by image example but, they neither focus on detection (i.e., discrimination between trademark and non-trademark images) nor do they perform retrievals on the Web by image content. This work handles both these issues. Higher quality results are obtained when more important (authoritative) Web pages are assigned higher ranking over less important pages. This work implements PicASHOW and WPicASHOW, two well established methods for image link analysis and retrieval on the Web. Compared with PicASHOW, WPicASHOW allows also for more sophisticated image queries such as queries by example image in addition to text queries.

A complete prototype Web retrieval system for the retrieval of logo and trademark images is also designed and implemented as part of this work. The system stores a crawl of the Web rich in image and text content and offers the framework for a realistic evaluation of many Web image retrieval methods including PicASHOW, WPicASHOW and the Vector Space Model. The experimental results demonstrate that WPicASHOW is far more effective than PicASHOW, which uses link information alone. Link analysis improved the quality of the results but not necessarily their accuracy (at least for data sets smaller than the Web). The analysis revealed that content relevance and searching for authoritative answers can be traded-off against each other: Giving higher ranking to important pages seems to reduce the accuracy of the results.

Future work includes experimentation with larger data sets and image types, more elaborate methods for logo

and trademark detection and matching, and more elaborate crawling methods for fetching pages relevant to the image type of the application (focused crawling).

Acknowledgements

This work was supported by project PYTHAGORAS of the Greek Secretariat for Research and Technology.

7. REFERENCES

[1] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepke, and S. Raghavan. Searching the Web. *ACM Trans. on Internet Technology*, 1(1):2–43, Aug. 2001.

[2] Y.A. Aslandongan and C.T. Yu. Evaluating Strategies and Systems for Content-Based Indexing of Person Images on the Web. In 8^{th} *Intern. Conf. on Multimedia*, pages 313–321, Marina del Rey, CA, 2000.

[3] K. Bharat, A. Broder, M. R. Henzinger, P. Kumar, and S. Venkatasubramanian. The Connectivity server: Fast access to Linkage Information on the Web. In *Proceedings of the 7th International World Wide Web Conference (WWW-7)*, pages 469–477, Brisbane, Australia, 1998.

[4] K. Bharat and M. R. Henzinger. Improved Algorithms for Topic Distillation in a Hyperlinked Environment. In *Proc. of SIGIR-98*, pages 104–111, Melbourne, 1998.

[5] A.D. Bimbo. *Visual Information Systems*, chapter 2. Morgan Kaufmann, Academic Press, 1999.

[6] T. Gevers and A.W.M. Smeulders. The PicToSeek WWW Image Search Engine. In *IEEE ICMS*, June 1999.

[7] J.-Hu and A. Bagga. Identifying Story and Preview Images in News Web Pages. In 7^{th} *Intern. Conf. on Document Analysis and Recognition (ICDAR'2003)*, pages 640–644, Edinburgh, Scotland, Aug. 2003.

[8] A. K. Jain and A. Vailaya. Shape-Based Retrieval: A Case Study With Trademark Image Databases. *Pattern Recognition*, 31(9):1369–1399, 1998.

[9] M.L. Kherfi, D. Ziou, and A. Bernardi. Image Retrieval from the World Wide Web: Issues, Techniques, and Systems. *ACM Comp. Surveys*, 36(1):35–67, March 2004.

[10] J. M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5):604–632, 1999.

[11] R. Lempel and A. Soffer. PicASHOW: Pictorial Authority Search by Hyperlinks on the Web. *ACM Trans. on Info. Systems*, 20(1):1–24, Jan. 2002.

[12] B. M. Mehtre, M. S. Kankanhalli, and W. F. Lee. Content-Based Image Retrieval using a Composite Color-Shape Approach. *Information Processing and Management*, 34(1):109–120, 1998.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Computer Systems Laboratory, Stanford Univ., CA, 1998.

[14] E.G.M. Petrakis, A. Diplaros, and E. Milios. Matching and Retrieval of Distorted and Occluded Shapes using Dynamic Programming. *IEEE Trans. on Pattern Analysis and Machine Intel.*, 24(11):1501–1516, Nov. 2002.

[15] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[16] Eds R. Baeza-Yates. *Modern Information Retrieval*. Addison Wesley, 1999.

[17] M. Tanase R.C. Veltkamp. Content-Based Image Retrieval Systems: A Survey. Technical Report UU-CS-2000-34, Department of Computing Science, Utrecht University, Oct. 2001. http://www.aa-lab.cs.uu.nl/cbirsurvey/cbir-survey.

[18] G. Salton, A. Wong, and C.-S. Yang. A Vector Space Model for Automatic Indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[19] H.-T. Shen, B.-Chin Ooi, and K.-Lee Tan. Giving Meanings to WWW Images. In 8^{th} *Intern. Conf. on Multimedia*, pages 39–47, Marina del Rey, CA, 2000.

[20] A. W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-Based Image Retrieval at the End of the Early Years. *IEEE Trans. on Pattern Analysis and Machine Intel.*, 22(11):1349–1380, Dec. 2000.

[21] J.R. Smith and S.-Fu Chang. Visually Searching the Web for Content. *IEEE Multimedia*, 4(3):12–20, July-Sept. 1997.

[22] M. Sonka, V. Hlavec, and R. Boyle. *Image Processing Analysis, and Machine Vision*, chapter 14. PWS Publishing, 1999.

[23] M. Sonka, V. Hlavec, and R. Boyle. *Image Processing Analysis, and Machine Vision*, chapter 6. PWS Publishing, 1999.

[24] L. Taycher, M.La Cascia, and S. Sclaroff. Image Digestion and Relevance Feedback in the ImageRover WWW Search Engine. In 2^{nd} *Intern. Conf. on Visual Information Systems*, pages 85–94, San Diego, Dec. 1997.

[25] E. Voutsakis, E. G.M. Petrakis, and E. Milios. IntelliSearch: Intelligent Search for Images and Text on the Web. In 3^{rd} *Intern. Conference on Image Analysis and Recognition (ICIAR 2006)*, pages 697–708, Povoa de Varzim, Portugal, Sept. 2006.

[26] E. Voutsakis, E.G.M. Petrakis, and E. Milios. Weighted link analysis for logo and trademark image retrieval on the web. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI2005)*, pages 581–585, Compiegne, France, Sept. 2005.

[27] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, chapter 4. Morgan Kaufmann, Academic Press, 2000.