Relevance Feedback Methods for Logo and Trademark Image Retrieval on the Web

Euripides G.M. Petrakis[†]
petrakis@intelligence.tuc.gr

Klaydios Kontis[†]
clauss13@softnet.tuc.gr

Epimenidis Voutsakis[†]
pimenas@softnet.tuc.gr

Evangelos E. Milios[‡]
eem@cs.dal.ca

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

ABSTRACT

Relevance feedback is the state-of-the-art approach for adjusting query results to the needs of the users. This work extends the existing framework of image retrieval with relevance feedback on the Web by incorporating text and image content into the search and feedback process. Some of the most powerful relevance feedback methods are implemented and tested on a fully automated Web retrieval system with more than 250,000 logo and trademark images. This evaluation demonstrates that term re-weighting based on text and image content is the most effective approach.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance feedback, Query formulation, Retrieval models, Search process*

General Terms

Performance, Experimentation, Algorithms

Keywords

Image retrieval, Relevance feedback, World Wide Web

1. INTRODUCTION

Effective image retrieval on the Web requires integration of text and image content information into the retrieval process [5]. A method is successful if it retrieves the images that the user expects to see in the answers with as few errors as possible. This is a highly subjective processes (i.e., the same results may be judged differently by different users).

Query uncertainly and user subjectivity may have a disastrous impact on the quality of the results. Query uncertainty depends on users level of expertise or familiarity with the system and system functions. Most commonly, users perceive image content in terms of high or semantic level concepts while, in the system, image content is represented in terms of low level image features (e.g., color, texture features). Consequently, users cannot express their information needs in queries or, even worst, there may exist a degree of uncertainty in queries as to what the users are really looking for.

Relevance feedback [16, 17] is the state-of-the-art approach for adjusting query results to the needs of the users. A common assumption is that there exists an ideal query (or matching method) that captures the information needs of the users. Relevance feedback attempts to guess the ideal query (or matching method) from answers that are initially obtained from the database. Typically, the users mark relevant (positive) or irrelevant (negative) examples among the retrieved answers, these examples are processed to form a new query which is combined with the original query and is resubmitted to the system. The process is repeated until convergence (i.e., the answers do not change). A categorization of methods includes:

Query point movement methods assuming that the ideal query is a point in a multi-dimensional space that the method approximates iteratively [8].

Term re-weighting methods that adjust the relative importance (weights) of terms in image representations [10, 3]. Terms that vary less in the set of positive examples are more important and should weigh more in retrievals. The inverse of the standard deviation is usually used for re-weighting the query terms.

Query expansion methods that attempt to guess an ideal query by adding new terms into the user's query [11, 2, 6].

Similarity adaptation methods that approximate the ideal matching method by substituting the system similarity (or distance) function with one that better captures the user's notion of similarity [15].

There are also approaches combining the above ideas. MindReader [3] combines query point movement and term re-weighting and handles correlations between attributes. Weight estimation is formulated as a minimization problem. MARS [9] is a prototype image retrieval system implementing a variation of the standard term re-weighting method. "iFind"[6] supports keyword-based image search along with queries by image example. The main idea behind this approach is that images which are similar to

SAC'06 April 23-27, 2006, Dijon, France

the same query represent similar semantics. Images are linked to semantics by applying data mining on user's feedback log [2]. A recent contribution [17] presents a critical survey of several relevance feedback approaches for image retrieval. This work neither presents an experimental evaluation nor does it show how to apply relevance feedback for queries combining text and image content.

The contributions of this work are summarized in the following:

- The existing framework of image retrieval with relevance feedback on the Web is extended to handle more sophisticated queries (e.g., queries by image example), by incorporating text and image content into the image retrieval and relevance feedback processes.

- Several relevance feedback methods are implemented and evaluated including, term re-weighting [10], query expansion [11] and similarity adaption methods [15]. Query point movement are restricted to vector representations and are not examined (in this work image content representations are not vectors). This evaluation demonstrates that term re-weighing combined with retrieval methods integrating text and image content is the most effective approach.

A complete and fully automated system for retrieval of logo and trademark images is used as a testbed for the evaluation of all candidate methods [14]. The system supports indexing and storage for Web pages, images, and information extracted from them (e.g., text, image features).

Image content description and image similarity are discussed in Sec. 2. The relevance feedback methods for logo and trademark image retrieval under consideration are discussed in Sec. 3. The image retrieval system used for the evaluations is presented in Sec. 4. Experimental results are presented in Sec. 5 followed by conclusions in Sec. 6.

2. IMAGE CONTENT REPRESENTATION

We choose the problem of logo and trademark images as a case study for the evaluation of relevance feedback methods. Retrieval of logo and trademarks is of significant commercial interest (e.g., patent offices or systems like ImageLock[1] provide detection services of unauthorized uses of logos and trademarks). This technology can greatly benefit from the proposed approach. Because images are not properly categorized on the Web, filters based on learning by decision trees for selecting Web pages with logo and trademark images are also proposed.

Existing approaches for retrieval of trademarks (e.g., [4, 7]) focus entirely on image content analysis and high precision answers to queries by image example but, neither focus on detection (i.e., discrimination between trademark and not trademark images) nor do they apply for retrievals on the Web. This work handles both these issues. The focus of this work is not on image feature extraction but, on improving the quality of retrievals by relevance feedback for a given and well appreciated set of features.

2.1 Text Features

Typically, images are described by text surrounding the images in the Web pages [11]. The following types of image descriptive text is derived based on the analysis of `html` formatting instructions:

Image Filename: The `URL` entry in the `src` field of the `img` formatting instruction.

[1]www.imagelock.com

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

Caption: A sentence that describes the image. It does not correspond to any `html` formatting instruction. It is is limited to 30 words before or after the reference to the image file.

Page Title: It is contained between the `TITLE` formatting instructions in the beginning of the document. It is optional.

Each part of the above description is syntactically analyzed and represented by an ordered list of stemmed terms appearing into it (the terms are taken in the same order as they appear in the text) [11]. The text similarity between a query Q and an image I is computed as

$$S^{text}(Q, I) = \sum_{i \,\in\, representation} w_i^{text} S_i^{text}(Q, I), \qquad (1)$$

where w_i^{text} are weights (inner weights) denoting the relative significance of the above lists. Each S_i component is computed as list similarity: The more common terms (in the same order) two term lists have in common, the more similar they are. Notice that [11] neither shows how to select good weights nor does it show how to handle image content in queries. This work handles both these issues.

2.2 Image Features

Because the same logo or trademark image may appear as color or grey scale image in different Web pages, color information is not useful in content representations. 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 [12]: 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). The histogram values are normalized by the 0-th component.

Moment Invariants [12]: A representation of 7 moment coefficients of the shape is computed from the area it occupies. It has been proven to be particularly effective in retrieval of logo and trademark images [4, 7].

The purpose of this type of representations is twofold:

Logo-Trademark Detection: A five-dimensional vector is formed: Each image is specified by the mean and variance of its intensity and energy spectra 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. Their feature vectors are fed into a decision-tree which is trained to detect logo and trademark images. For each image the decision tree computes an estimate of its likelihood of being logo or trademark or "Logo-Trademark Probability".

Logo-Trademark Similarity: The image similarity between a query image Q and an image I is computed as

$$S^{image}(Q, I) = \sum_{i \,\in\, representation} w_i^{image} S_i^{image}(Q, I), \qquad (2)$$

where w_i^{image} are weights (inner weights) denoting the relative significance of the above types of image content representations. The computation of each S_i component depends on feature type: The similarity between histograms is computed by their intersection whereas the similarity between moment invariants is computed by subtracting the Euclidean vector distance from its maximum value.

2.3 Image Similarity

To answer queries combining text and image example, the similarity between a query Q and a Web image I is computed as

$$S(Q, I) = W^{image} S^{image}(Q, I) + W^{text} S^{text}(Q, I), \quad (3)$$

where W^{text} and W^{image} are weights (outer weights) denoting the relative significance of image and text descriptions. All measures above are normalized to lie in the interval [0,1].

3. METHODOLOGY AND BACKGROUND

The inner and outer weights of Eq. 1, Eq. 2 and Eq. 3 place different emphasis on different features or representations respectively and can be used to adapt the query results to user's preferences. Typically, the weights are user defined. However, weight definition is beyond the understanding of most users. Relevance feedback is employed to estimate good weight values. Query expansion, term re-weighting and similarity adaptation methods are considered as representatives of most important categories of methods. Query point movement methods assume vector representations and cannot be applied. In the following the basic steps of each method are discussed. The same steps are applied iteratively until convergence (i.e., the results of the retrieval method do not change). Initial results are obtained by applying either Eq. 1 (for text queries) or Eq. 3 (for queries combining text with image example). All weights are initialized to 1.

3.1 Query Expansion

The query is expanded with new terms obtained from positive examples. Two methods are evaluated. These methods work only with text.

Accumulation [11]: The most relevant image is selected from the answers and its text representation (i.e., a list of descriptive terms) is extracted. The query is matched with each term in this representation. A new query is formed by merging the query representation with the most similar terms of the most relevant image.

Integration and Differentiation [11]: Relevant and irrelevant images are selected from the answers. From each relevant image, its text representation (i.e., list of descriptive terms) is extracted and matched with the query. The most similar terms are combined to form a new "positive query". Similarly, the most dissimilar (to the query) terms are extracted from all irrelevant answers and combined to form a "negative query". The positive query is applied. Images which are more similar to the negative query rather than to the positive query are removed from the the answer.

3.2 Term Re-Weighting

Term re-weighting works by adjusting the relative importance of query terms [10]. In this work the method is extended to accommodate for the definition of image similarity by text and image content as follows.

Let R be the set of the N_R most similar images (in this work $N_R = 30$). A relevance score taking values -3 (for highly non-relevant answers) through 3 (for highly relevant answers) is assigned to each answer in R (neutral or no-opinion answers take score 0). R also denotes the query results at the beginning of each feedback cycle.

The outer weights W^j ($j \in \{text, image\}$) are dynamically updated during each feedback cycle: The database is queried by each S^j separately (using either Eq. 1 or Eq. 2) and its answer set R^j is sorted by similarity. The weights are then updated according to the following formula

$$W^j = \begin{cases} W^j + score_I & \text{if } I \in R, \\ W^j + 0 & \text{otherwise}, \end{cases} \quad (4)$$

where $score_I$ is the score assigned to image I in R. Initially all $W^j = 0$. After iterating over the images in each R^j all weights W_i^j are normalized by $W_{total}^j = \sum_{I \in R^j} W^j$. Negative weights are set to 0.

The inner weights $w_i^j (j \in \{text, visual\})$ for each term i of the text or image representation are also dynamically updated using the set R' of relevant answers in R ($R' \subset R$): The smaller the variance of each S_i^j the larger the significance of the i-th term (and the reverse). Therefore, $w_i^j = 1/\sigma_i^j$, where σ_i^j is the variance of the i-th feature in the j-th representation. Each weight is normalized by $w_{total}^j = \sum_{I \in R'} w_i^j$.

3.3 Similarity Adaptation

Falcon [15] estimates an ideal distance function $\mathcal{D}_\mathcal{G}$ that retrieves the best results. Initially, Falcon searches the database using $d(Q, I) = 1 - S(Q, I)$ as distance function and the user adds positive examples to a set $\mathcal{G}$ (initially empty). During a feedback cycle, Falcon searches the database again using a new distance function $\mathcal{D}_\mathcal{G}$ while the user adds new positive examples to $\mathcal{G}$. The distance between the query Q and a Web image I is computed as the distance of I from the current members of $\mathcal{G}$. Falcon estimates $\mathcal{D}_\mathcal{G}$ iteratively as follows

$$\mathcal{D}_\mathcal{G}(I) = \begin{cases} 0 & \text{if } \exists i : d(g_i, I) = 0 \\ \left(\frac{1}{k} \sum_{i=1}^{k} d(g_i, I)^\alpha \right)^{1/\alpha} & \text{otherwise}, \end{cases} \quad (5)$$

where k is the number of positive examples in $\mathcal{G}$, g_i is a member of $\mathcal{G}$ and α is a user defined constant (in this work $\alpha = -5$).

4. WEB RETRIEVAL SYSTEM

All methods methods are evaluated using a prototype retrieval system for images in Web pages [14] as a testbed. The system is available on the Web [2]. Fig. 1 illustrates the architecture of the system. The system consists of several modules, the most important of them being the following:

Crawler module: Implemented based upon Larbin [3], the crawler assembled locally a collection of Web pages from which more than 250,000 pages contain logo and trademark 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 [4] to 20 queries on various topics (e.g., topics related to Linux and software products). The crawler worked recursively in breadth-first order and visited pages up to a depth of 5 links from each origin.

[2]http://www.ece.tuc.gr/intellisearch
[3]http://larbin.sourceforge.net
[4]http://www.google.com/imghp

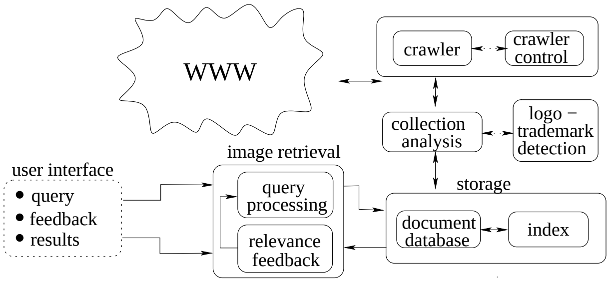

Figure 1: Web System Architecture.

Collection Analysis module: The content of crawled pages is analyzed. Text, images, link information and information for pages that belong to the same site is extracted. For each image, its text description is extracted.

Storage module: Implements storage structures and indices providing fast access to Web pages and information extracted from them (i.e., text, image descriptions and link information).

Image Retrieval module: Implements the above search and feedback methods. Queries are issued by keywords (or free text) or by combination of keywords and image example.

Logo-Trademark Detection module: Detects images with high probability of being logos or trademarks. Only images with probability higher than 0.5 are stored in the database and used in retrievals.

The database is implemented in BerkeleyDB [5]. Two inverted files implement the connectivity server [1] and provide fast access to linkage information between pages (backward and forward links). Two inverted files associate terms with their intra and inter document frequencies and allow for fast computation of term vectors.

The system is designed to support queries by image content for logo and trademark images on the Web. The problem of image retrieval on the Web is treated as one of retrieval by text or combined text and image features as described in Sec. 2.

5. EXPERIMENTS

The methods in Sec. 3 are implemented and evaluated. For the evaluations, 20 queries are created on topics related to Linux and software. The results are obtained after 2 feedback iterations (all methods converged after 2 feedback iterations). The evaluation is based on human relevance judgments by 4 independent referees. Each referee evaluated a subset of 5 queries for all methods. Each query retrieves the best 30 answers. For each method the average precision and recall over 20 queries is computed. A method is better than another if it achieves higher precision and recall.

Due to the large size of the data set, it is practically impossible to compare every query with each image in the database. 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. This is a valid sampling method known as *"pooling method"* [13]. This method allows for relative judgements (e.g., method A retrieves 10% more relevant answers than method B) but does not allow for absolute judgements (e.g., method A retrieved 10% of the total relevant answers).

[5]http://www.sleepycat.com

5.1 Text Queries

All queries specified the term "logo". A Web image is similar to the query if it is on the same topic with the query. Query "Linux logo" may retrieve the logo image of any Linux distribution (e.g., "Debian Linux"). The "naive" method corresponds to database searching using Eq. 1.

Table 1: Performance of relevance feedback methods for text queries.

Method	Precision	Recall
Naive Search using Eq. 1	0.48	0.63
Accumulation [11]	0.44	0.57
Integration and Differentiation [11]	0.44	0.55
Term Re-Weighting [10]	0.56	0.72
Falcon [15]	0.52	0.67

Table 5.1 illustrates the precision/recall values of all methods. The precision/recall values of naive search (before feedback) are also shown. Term re-weighting is obviously the best method followed by Falcon. Both methods improved naive search. Query expansion methods failed to improve the performance of the naive method. A closer look into the results revealed that both methods expanded the query with noisy terms (thus leading to topic drift) while, differentiation removed correct terms from the query (in some cases).

Term re-weighting maintains the same good performance even when feedback is provided for the first 15 answers only. This is an important advantage of the method allowing for smaller iteration cycles and therefore for faster retrievals with less effort. Falcon demonstrated exactly the opposite behavior: Requires the maximum possible feedback in order to approximate a good distance function.

5.2 Text Queries with Image Example

Each keyword query is augmented by an example logo image. An answer is similar to the query only if it is on the same topic with the query and also contains an image similar to the query image (e.g., query "Linux logo" with the penguin logo is similar to answers on Linux showing a Linux penguin logo).

Table 2: Performance of relevance feedback methods for text queries with image example.

Method	Precision	Recall
Naive Search using Eq. 3	0.52	0.46
Term Re-Weighting [10]	0.60	0.65
Falcon [15]	0.54	0.44

Query expansion methods work only for text and are not considered. The naive method corresponds to database searching using Eq. 3. Table 5.2 illustrates that term re-weighting is the most effective method achieving better precision and significantly better recall than naive search, even for smaller iteration cycles (feedback is provided for the first 15 answers). Notice that the method is more effective for such complex queries than it is for text queries. Falcon failed to improve naive search indicating a weakness to approximate complex similarity functions involving both text and image features.

6. CONCLUSIONS

This work extends the existing framework for image retrieval with relevance feedback on the Web to support queries by text and example image. The evaluation is based on a prototype Web retrieval system for logo and trademark images which stores a crawl of the web and offers the framework for a realistic evaluation of relevance feedback methods. The results demonstrate, that term re-weighting is the most effective approach for all query types. Term re-weighting allows also for much smaller iteration cycles (and therefore for faster retrieval with less users effort) while maintaining good performance. All methods converge very fast (i.e., after two iteration cycles). Future work includes extension of term re-weighting methods to work on image meta-data (e.g., user's log information) and combination of relevance feedback methods with link-analysis methods for assigning higher ranking to good quality images (over other relevant images) while preserving user's preferences.

7. REFERENCES

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

[2] Z. Chen, L. Wenyin, F. Zhang, M. Li, and H.-J. Zhang. Web Mining for Web Image Retrieval. *Journal of the American Society of Information Science*, 52(10):831–839, 2001.

[3] Y. Ishikawa, R. Subramanya, and C. Faloutsos. Mindreader: Query Databases Through Multiple Examples. In *Proc. of the 24th VLDB Conference*, pages 218–227, New York. USA, 1998.

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

[5] M. 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.

[6] Y. Lu, C. Hu, X. Zhu, H.-J. Zhang, and Q. Yang. A Unified Framework for Semantic and Feature Based Relevance Feedback in Image Retrieval Systems. In *Proc. ACM Multimedia*, pages 31–37, Los Angeles CA, USA, 2000.

[7] 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.

[8] J. Rocchio. Relevance Feedback in Information Retrieval. In G. Salton, editor, *The SMART Retrieval System - Experiments in Automatic Document Processing*, pages 313–323. Prentice Hall, Englewood Cliffs, 1971.

[9] Y. Rui, T. Huang, and S. Mehrotra. Content-Based Image Retrieval with Relevance Feedback in MARS. In *Proc. IEEE Int. Conf. on Image Processing*, pages 515–518, Santa Barbara, CA, Oct. 1997.

[10] Y. Rui, T.-S. Huang, M. Ortega, and S. Mechrota. Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval. *IEEE Trans. on Circ. and Syst. for Video Technology*, 8(5):644–655, Sept. 1998.

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

[12] M. Sonka, V. Hlavec, and R. Boyle. *Image Processing Analysis, and Machine Vision*, chapter 6 & 14. PWS Publishing, 1999.

[13] E. Voorhees and D. Harmann. Overview of the Seventh Text REtrieval Conference (TREC-7). In *NIST Special Publication 500-242: The Seventh Text REtrieval Conference (TREC-7)*, pages 1–23, 1998.

[14] E. Voutsakis, E. Petrakis, and E. Milios. Weighted Link Analysis for Logo and Trademark Image Retrieval on the Web. In *Proc. IEEE/WIC/ACM Intern. Conf. on Web Intelligence (WI2005)*, pages 581–585, Compiegne, France, Sept. 2005.

[15] L. Wu, K. Sycara, T. Payne, and C. Faloutsos. FALCON: Feedback Adaptive Loop for Content-Based Retrieval. In *Proc. 26th VLDB Conf.*, pages 297–306, Cairo, Egypt, Sept. 2000.

[16] H.-J. Zhang, Z. Chen, W.-Y. Liu, and M. Li. Relevance Feedback in Content-Based Image Search. In *Proc. 12th Intern. Conf. on New Information Technology (NIT)*, pages 29–31, Beijing, China, Aug. 2003.

[17] X.-S. Zhou and T. Huang. Relevance Feedback in Image Retrieval: A Comparative Study. *Multimedia Systems*, 8(6):536–544, April 2003.