DESIGN OF A WEB-BASED INTERFACE FOR IMAGE RETRIEVAL SYSTEMS

Liam M. Mayron and Oge Marques
Department of Computer Science and Engineering, Florida Atlantic University, 777 Glades Rd., Boca Raton, FL, USA 33431

Keywords: Image retrieval, image annotation, web-based interface.

Abstract: In this work we present a new interface for image retrieval system that highlights the potential of a real-time, Web-based application. This system, the Perceptually-Relevant Image Search Machine (PRISM), combines the capabilities of content-based, content-free, and semantic annotation-based image retrieval. Each of the aforementioned image retrieval methods has its own strengths, weaknesses, and impracticalities. It is our hope that the PRISM interface, by enabling the simultaneous expression of all three methods, will lead to more robust image retrieval systems.

1 INTRODUCTION

Image retrieval systems have long been in development, but limitations still exist. Many implementations remain prototypes and fail to address the fundamental issues facing image retrieval, namely the semantic gap and sensory gap.

This work outlines the development of a new interface for image retrieval. It is motivated by the identification of a shortcoming of existing image retrieval systems, the *interface gap*. We define the interface gap as the loss of expressiveness due to an image retrieval system's interface. To address this we present the Perceptually-Relevant Image Search Machine (PRISM), first demonstrated in (Mayron et al., 2006). PRISM was developed with many capabilities normally not available in combination in image retrieval systems, notably the ability to combine content-based, content-free, and semantic annotation information into a single query and the ability to accommodate multiple concurrent and discontinuous user sessions.

This paper is organized as follows. Section 2 presents relevant background information. Section 3 discusses the functionality and implementation of the PRISM interface. Section 4 presents applications of the new interface. Finally, Section 5 concludes the paper.

2 BACKGROUND

There are two well-known and unresolved issues in image retrieval: the *semantic gap* and the *sensory gap* (Smeulders et al., 2000).

The semantic gap is the difference that exists between the user's interpretation of an image and what can be determined based on the physical properties of that image (Smeulders et al., 2000). For example, a human might see a picture of a car and identify it as such based on past experiences and memories. This is a complex task that is extremely difficult to model computationally, where only modifications of pixel values are available. In summary, a user's description and a computer's description of the same visual data are likely to differ significantly.

The sensory gap is brought about by the translation of our 3D world onto a flat, two-dimensional, discrete array of pixel values (Smeulders et al., 2000). Information is lost during this translation which is difficult to reproduce. Making sense of a 3D environment from a single 2D projection is a challenge for image retrieval systems (when photographs and naturally occurring scenes are considered, such as in this work).

We have identified a third gap for image retrieval, the *interface gap* (see Figure 1). The process of translating the user's motivation into physical actions (typ-

M. Mayron L. and Marques O. (2007).
DESIGN OF A WEB-BASED INTERFACE FOR IMAGE RETRIEVAL SYSTEMS.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Web Interfaces and Applications*, pages 328-333
DOI: 10.5220/0001288003280333

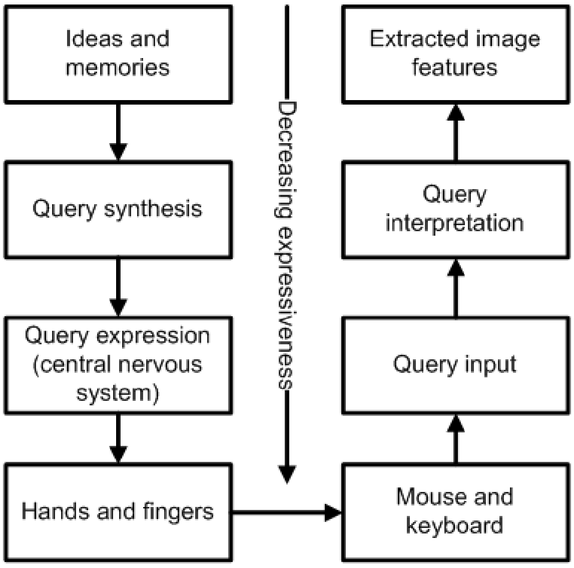

Figure 1: The interface gap.

ically mouse clicks) and then reconstructing those actions results in a loss of expressiveness. Once expressiveness is lost it is very difficult to interpolate, resulting in the interface gap. Our motivation for creating PRISM was to allow for more expressive queries, thus narrowing the interface gap.

3 DESCRIPTION

PRISM introduces a new type of interface for image retrieval systems that combines content-based, content-free, and semantic annotation-based query capabilities. Its functionality and implementation details are presented in this section.

3.1 Functional Description

A user must create a unique profile before using PRISM. It is also possible to use a guest account. Each user account is protected with a password. The purpose of a unique account is to allow user activity to be recorded individually. This information can then be aggregated and used to improve the image retrieval (for example, to aid in content-free analysis). In this case it is possible to be able to suspend the session and resume later. This is helpful if a user wants to use PRISM solely for image organization they may not be able to complete the task in a single session. Finally, inferences can be gained by observing how a user's activity progresses across multiple sessions.

Once the user has created an account, or if they are resuming a previous session, they may sign in using the credentials they established.

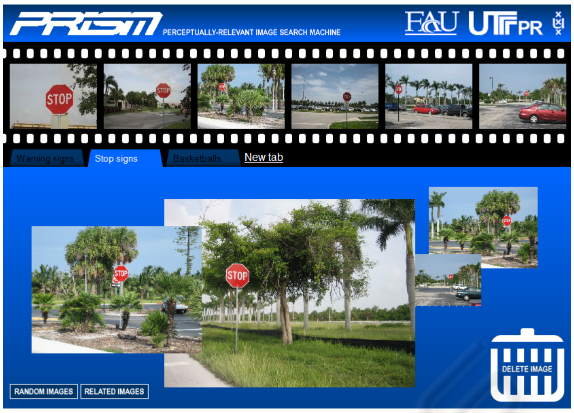

Figure 2: The PRISM interface.

The initial, default view of the PRISM desktop is divided into three distinct horizontal partitions (Figure 2).

The top of the screen provides information on the current session, access to documentation, and the ability to save and terminate the session.

The middle section, referred to as the "filmstrip", is an important part of the PRISM interface. The metaphor of a filmstrip is useful in describing the function of this portion of the interface and is reinforced with visual elements. The filmstrip is the only source of new images in PRISM. The user drags images from the filmstrip into anywhere on the main content area. An image may be deleted from the filmstrip by dragging it to the trash can icon in the lower-right corner of the screen. When an image is removed from the filmstrip the images behind it move up and the vacant space is immediately filled with a new image. Thus, the filmstrip is an always-full collection of images for the user to assess.

The third section of the PRISM interface is the largest – the tabbed content area (also referred to as the workspace). It is within this area that images are organized. These organized images form the basis for the content-based, content-free, and semantic queries that may be posed. The workspace is crowned by a variable number of tabs. It is significant that these tabs can be individually labeled – this information can be used in semantic retrieval. Tabs have become a popular interface construct, and for good reason. They expand and segment the functional area while occupying a minimal amount of space. In PRISM, tabs are used to organize individual groups of images, expanding the total available work area, while avoiding overwhelming the user with too many images visible at once.

A small number of controls have been placed at the bottom of the workspace. To the left are two buttons labeled “Random Images” and “Related Images”. Both of these buttons will empty the filmstrip and replace it with either random images from the image database, or related images. As mentioned earlier, PRISM can accommodate queries that are content-based, content-free, or semantic annotation-based. To the right is the trash can icon. For consistency, it was decided that dragging images to this area would delete images, rather than using separate buttons on each image for deletion. Images can be deleted directly from the filmstrip, or from the workspace after they have been placed.

The workspace is used for arranging images. Images can be placed anywhere and moved to new locations after their initial assignment by clicking and dragging. This action was inspired by the method one might use to arrange a shoe box full of images. In PRISM the intention is for the user to place related images closer together. It can be inferred that images that are placed close together within a tabs and, to a lesser degree, images that share a tab, are related. This functionality enables content-free queries. If, across many users, the same images occur together their likelihood of being related increases. This can be judged regardless of content (hence, content-free).

Within the workspace images can also be scaled larger or smaller. It is our intention that users will make more relevant and important images larger, and vice versa. In content-based queries larger images can be given more weight. This capability replaces the relevance feedback found in other systems. Additionally, since the smaller and images is, the more difficult it is to view details, image size can be used to distinguish determine is a user is intrigued by the content of the whole image (the *gist of a scene* (Oliva, 2005)) or a specific region of interest. Scaling is accomplished by clicking on the “+” and “-” icons that appear when the cursor is moved over an image. Placing these controls in an overlay that is only visible when needed reduces the complexity of the interface.

Finally, the workspace allows the annotation of individual images (as well as the tabs the images belong to). Previously annotated images (by the same user or other users) can be recalled based on an analysis of the annotations in the current workspace. Images are annotated by changing the text in the same overlay that appears to resize images.

In summary, the functionality of PRISM is simply described and demonstrated, yet enables expressive queries without requiring knowledge of the inner workings of the system.

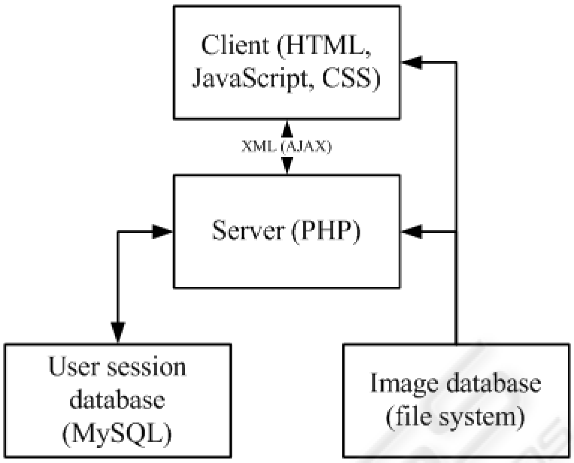
Client (HTML, JavaScript, CSS)
XML (AJAX)
Server (PHP)
User session database (MySQL)
Image database (file system)

Figure 3: General architecture.

3.2 Implementation

The PRISM interface was implemented using contemporary technologies and design methodologies. From the beginning, it was intended that PRISM would be a web-based applications. Ultimately, this provides the broadest platform. Accessibility is of paramount importance to any interface, and designing for widely-used Web browsers helps satisfy this need. The client-server architecture allows query processing to occur on the remote server, freeing the client of processing and storage requirements. The image database communicates independently with both the client and server, providing images as requested. The client-server architecture also helped fulfill the multi-user requirement of the implementation.

Figure 3 shows the general architecture of the system. The client and server communicate using AJAX (more generally, XML). The server has access to the MySQL database for storing and retrieving user sessions. Both the server and client have the ability to retrieve images from the image database.

On the client-side JavaScript, CSS, and the DOM (Document Object Model) realize the interfaces functionality (collectively known as DHTML). Ajax (Asynchronous JavaScript and XML) is used throughout to communicate with the PHP-based server, MySQL relational database, and the image database. Additional functionality was incorporated from the popular Script.aculo.us (script.aculo.us, 2006) and Xajax (xajax, 2006) libraries.

The client instantiates each image as an object contained within a standard HTML division (div) with a unique ID to facilitate access and tracking through the DOM. CSS is used heavily throughout to template common elements and to reduce the amount of code generated by each new image. DHTML is used to move images and communicate with the server using the XMLHttpRequest Object (Ajax). Image position, size, and annotation is stored in JavaScript objects until it is committed to the server's database.

DHTML has been widely used for several years and is well-known. However, Ajax (Asynchronous JavaScript and XML) is relatively new. Ajax is a powerful method of communicating with a remote server. It an implementation of a remote procedure call using XML as the common data format. In PRISM it enables many relatively complex operations to occur in realtime without refreshing the entire view (which would consume a large amount of bandwidth and be quite inefficient). It also provides a structured way to transfer background data (that is not part of the display). Essentially, Ajax allows a web-based application to behave like a realtime, event-driven application (and be written like one as well). This is a fundamental and significant departure from web-based application architecture of the past.

The server was written in PHP and obtains data from a MySQL database. The back-end is nearly entirely event-driven except for the necessary initialization code. The Xajax library is used extensively to facilitate communication between the client and server. Indeed, it makes the implementation of many functions trivial. All it must do in this case is assign no content to the container object, in effect clearing it.

User information is stored in the MySQL database. Accesses to the database are kept to the absolute minimum. One access is made when the user connects to the system. When (and only when) the user leaves their session is written to the database.

4 APPLICATIONS

Several applications of the PRISM interface are presented in this section.

4.1 Content-based Image Retrieval

Content-based image retrieval (CBIR) determines which images to retrieve based on physical (pixel-based) properties. Color, intensity, and texture are common similarity measures, although others exist.

Similarity is determined by the distance of the retrieved images to the query. There are a variety of possible query types in CBIR systems that include, but are not limited to, interactive browsing, visual sketch (a drawing of the intended target), query-by-example, and query-by-specification of visual features (Marques and Furht, 2002). In many systems *relevance feedback*, the process of having the user refine results, is employed to improve results.

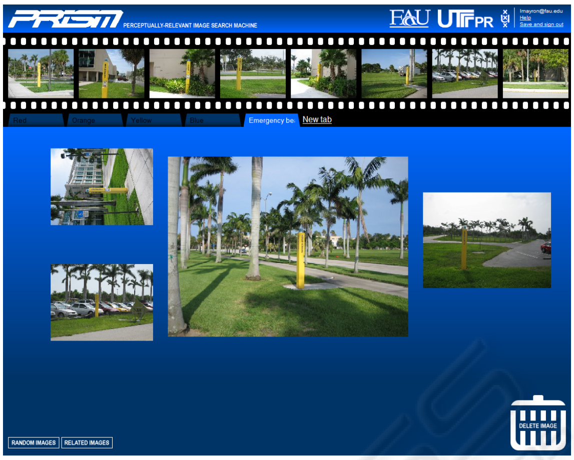

Figure 4: A representative content-based query.

A query method that has not yet been explored in detail is *query-by-multiple-example*. In this query method multiple example images are provided. The is the method we have implemented in PRISM (Figure 4). In our solution the user is able to select as many images as they desire to compose their initial query. Then, instead of a good-bad-neutral classifier for relevance feedback (as is very common), images may be scaled to indicate relevance. The larger a use makes an image, the more relevant it is considered to be, and vice-versa. The result is a query that is composed of multiple images, with each image being associated with its own relevance. A key benefit of this new interface is that, despite the detailed and expressive query that is generated, the user does not need expert knowledge of the system and can compose the query with minimal instruction.

4.2 Content-free Image Retrieval

Content-free image retrieval (CFIR) is a newer approach than CBIR. Rather than analyzing the specific properties of images, it relies on past information regarding the relations between images (Liu and Chen, 2006). In a CFIR system a user may associate related images together. A query will judge image similarity based on the history of the images appearing together.

PRISM was also designed to allow for content-free queries (Figure 5). Images can be arranged together as the user desired in order to express their relation. Because content-free retrieval requires (and is refined by) past history, the ability to pause, save, and resume sessions was introduced, as well as support for multiple, concurrent sessions. Multiuser scenarios

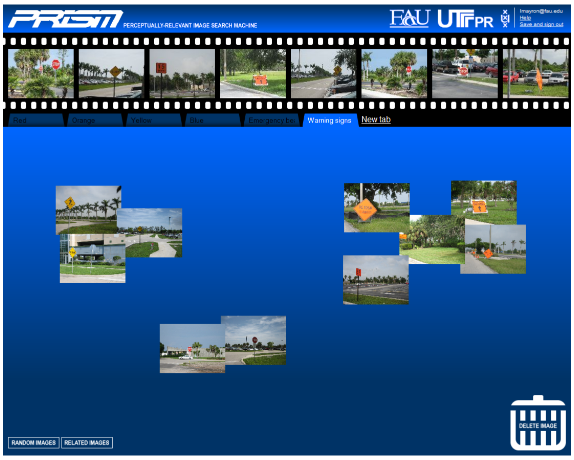

Figure 5: A representative content-free query.

are not typically considered by image retrieval applications – the general focus is on the experience of the individual user. In the case of PRISM, information from other users can be used in a content-free way to improve results. Being web-based greatly reduces the barriers for using the image retrieval system.

In PRISM, content-free relevance is established at several levels. At the broadest level, images placed on the same tab are related. At a finer level of granularity, the organization of images within a tab can be inspected. PRISM allows for individual clusters of images to be created. Images may also overlap to indicate pronounced inter-image relevance.

Through simple actions the user is able to quickly generate a complex set of content-free relationships.

4.3 Semantic Annotation

Semantic annotation is certainly among the most powerful ways to retrieve information. Its major inconvenience, however, is that it typically requires that the annotation be generated by human users. In many applications, including image retrieval, this is not a practical solution, which is why so much research has been performed along automatic and semi-automatic retrieval. Still, the PRISM interface does accommodate the incorporation of annotating an image database and retrieving images based on their semantic meaning.

Annotation is added to images by human users based on the semantic meaning they perceive (Marques and Barman, 2003). While an image of a playground may be interpreted by its physical properties in a CBIR system (e.g. blue, brown, and green colors), this does not necessarily reflex the semantic meaning of the image (which may be, in this case, a playground, children, outside, or childhood). It would be very difficult to specify a set of physical properties that would retrieve a set of images of a playground, despite the common semantic meaning.

In this case, PRISM allows both tabs and individual images to be annotated (Figure 6). In Figure 6 the semantic query is “objects on tables”. In this case, the retrieved images have been previously annotated and retrieved regardless of their physical similarity. The annotation of tabs allows for broader (category-level) information to be appended to a group of images. Then, individual images may be annotated with specific information. While it is not practical or desirable for a single user to annotate thousands of images, because PRISM supports multiple users semantic annotation is a relevant addition to its functionality.

4.4 Future Work

Because the PRISM interface allows for these queries to be performed not only independently, but simultaneously as well, new possibilities emerge. We are eager to investigate the new types of queries that are made possible by combining content-based information, content-free information, and semantic annotation.

We anticipate that the combination of these three broad methods of retrieval will result in a more robust query that is able to compensate for the shortcomings of individual methods. For example, both content-free retrieval and retrieval based on semantic annotation require a volume of existing, human-generated information to be practical. In a new system, content-based information could be used in the absence of such information. In another example, content-free information could be used to help determine in a more accurate way, which are the common features a user wished to base their retrieval on.

We encourage research in the field of image retrieval to investigate the rich, new possible queries that are enabled by the presented interface.

5 CONCLUSION

The PRISM interface enables a unique combination of content-based image retrieval, content-free image retrieval, and image retrieval based on semantic annotation. Each method of retrieving images has its own strengths and limitations. By creating a new interface for image retrieval that allows, in a simple and intuitive fashion, the combination of these three retrieval methods we hope to increase the expressiveness afforded to the user, thus narrowing the interface gap.

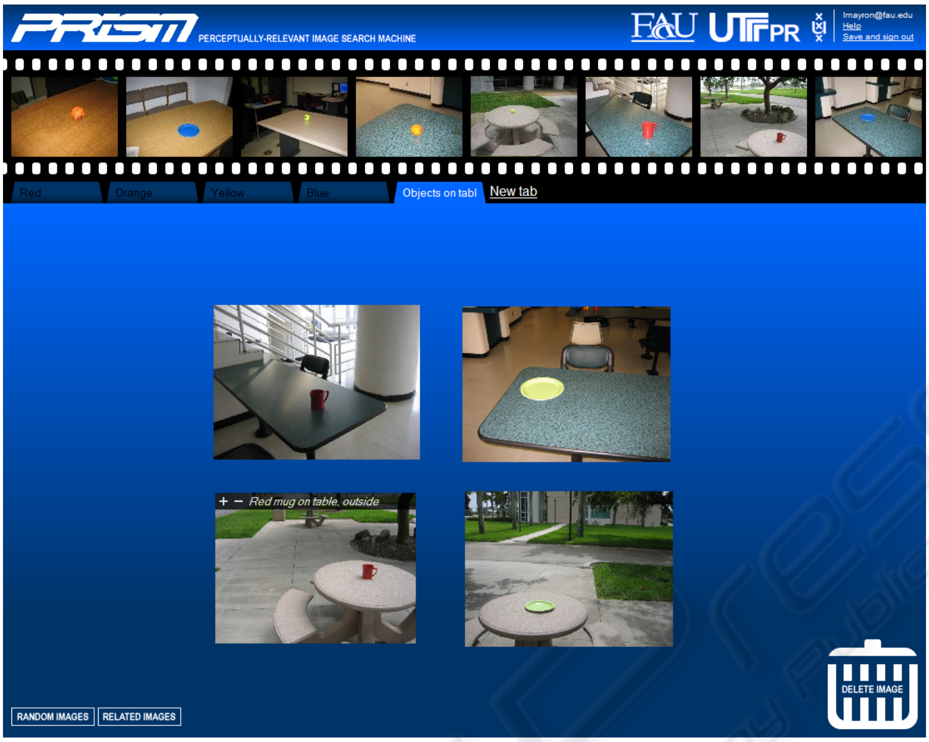

Figure 6: A representative query based on semantic annotation.

The new interface was realized using current web-based technologies. The benefits of these tools are clearly apparent, particularly in making the application portable and widely accessible.

The interaction of the query elements afforded by the PRISM interface will be the subject of our future work. Ultimately, we intend to create a complete, versatile, yet simple-to-use image retrieval system based on perceptually-sound principles.

ACKNOWLEDGEMENTS

This research was sponsored by the Office of Naval Research (ONR) under the Center for Coastline Security Technology grant N00014-05-C-0031.

REFERENCES

Liu, D. and Chen, T. (2006). Content-free image retrieval using bayesian product rule. In *IEEE International Conference on Multimedia & Expo*.

Marques, O. and Barman, N. (2003). Semi-automatic semantic annotation of images using machine learning techniques. In *International Semantic Web Conference*, pages 550–565.

Marques, O. and Furht, B. (2002). *Content-Based Image and Video Retrieval*. Kluwer Academic Publishers, Boston, MA.

Mayron, L. M., Borba, G. B., Nedovic, V., Marques, O., and Gamba, H. R. (2006). A forward-looking user interface for cbir and cfir systems. In *IEEE International Symposium on Multimedia (ISM2006)*, San Diego, CA, USA.

Oliva, A. (2005). Gist of a scene. In Itti, L., Rees, G., and Tsotsos, J., editors, *Neurobiology of Attention*, chapter 41. Academic Press, Elsevier.

script.aculo.us (2006). script.aculo.us - web 2.0 javascript. http://script.aculo.us.

Smeulders, A., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Trans. on PAMI*, 22(12):1349–1380.

xajax (2006). xajax php class library - the easiest way to develop asynchronous ajax applications with php. http://www.xajaxproject.org.