Journal of Visual Languages and Computing (1999) 10, 585–606
Article No. jvlc.1999.0147, available online at http://www.idealibrary.com on IDEAL®

Automatically Determining Semantics for World Wide Web Multimedia Information Retrieval

SOUGATA MUKHERJEA* AND JUNGHOO CHO[†] [‡]

**C&C Research Laboratories, NEC USA Inc., 110 Rio Robles, San Jose, CA 95134, U.S.A., e-mail: sougata@ccrl.sj.nec.com and [†] Department of Computer Science, Stanford University, U.S.A., e-mail: cho@cs.stanford.edu*

Accepted 18 August 1999

Search engines are useful because they allow the user to find information of interest from the World Wide Web (WWW). However, most of the popular search engines today are textual; they do not allow the user to find images from the web. For effective retrieval, determining the semantics of the images is essential. In this paper, we describe the problems in determining the semantics of images on the WWW and the approach of AMORE, a WWW search engine that we have developed. AMORE's techniques can be extended to other media like audio and video. We explain how we assign keywords to the images based on HTML pages and the method to determine similar images based on the assigned text. We also discuss some statistics showing the effectiveness of our technique. Finally, we present the visual interface of AMORE with the help of several retrieval scenarios.

Keywords: World Wide Web, image search, semantic similarity, HTML parsing, visual interface

1. Introduction

WITH THE EXPLOSIVE GROWTH OF INFORMATION that is available through the World Wide Web (WWW), it is becoming increasingly difficult for the users to find the information of interest. As most web pages have images, effective image search engines for the WWW need to be developed.

There are two major ways to search for an image. The user can specify an image and the search engine can retrieve images similar to it. The user can also specify keywords and all images relevant to the user-specified keywords can be retrieved. Over the last two years we have developed an image search engine called the *Advanced Multimedia Oriented Retrieval Engine* (*AMORE*) [1] (http://www.ccrl.com/amore) that allows the retrieval of WWW images using both the techniques. The user can specify keywords to retrieve relevant images or can specify an image to retrieve similar images.

For retrieving images by keywords we have to determine the meaning of the image. Obviously this is not very easy. The best approach will be to assign several keywords to

[‡] This work was performed when the author visited NEC.

1045-926X/99/120585+22 $30.00/0

an image to specify the meaning. Manually assigning keywords to images will give the best result but is not feasible for a large collection of images. Alternatively, we can use the surrounding text of web images as their keywords. Unfortunately, unlike written material, most HTML documents do not have an explicit caption. Therefore, we need to parse the HTML source file and only keywords ‘near’ an image should be assigned to it. However, because the HTML page can be structured in various ways, the ‘nearness’ is not easy to determine. For example, if the images are in a table, the keywords relevant to an image may not be physically near the image in the HTML source file. Thus, we require several heuristics to determine the keywords relevant to an image. Fortunately, these heuristics can be also applied to retrieve other media like video and audio from the web.

Once the keywords are assigned to the image, the user may specify keywords to retrieve relevant images. However, user studies with AMORE has shown that people also want to click on an image to find similar images. This kind of ‘*search for more like this one*’ is also popular for text search and is used in some WWW text search engines like Excite (http://www.excite.com). Especially for image searching, it is sometimes very difficult for the user to specify the kind of images she wants only by keywords.

The similarity of two images can be determined in two ways: *visually* and *semantically*. Visual similarity can be determined by image characteristics like shape, color and texture using image processing techniques. In AMORE, we use the *Content-oriented Image Retrieval* [2] library for this purpose. When the user wants to find images similar to a red car, COIR can retrieve pictures of other red cars. However, it may also be possible that the user is not interested in pictures of red cars but pictures of other cars having the similar manufacturer and model. Finding semantically similar images is useful in this case.

Since visual similarity does not consider the meaning of the images, a picture of a figure skater may be visually similar to the picture of an ice hockey player (because of the white background and similar shape), but it may not be meaningful for the user. To overcome this problem, AMORE allows the user to combine keyword and image similarity search. Thus, the user can integrate the visual similarity search of an ice hockey player picture with the keywords ‘ice hockey’. Although the integrated search retrieves very relevant images, unfortunately, an evaluation of AMORE’s access logs has shown that integrated search is not as popular as keyword or image similarity search. Naive WWW users do not understand the concept of integrated search. Therefore, automatically integrating semantic and visual similarity search may be more user-friendly.

For finding semantically similar images, we can assume that if two images have many common keywords assigned then they are similar. However, this simple approach has two drawbacks:

- Obviously, not all the keywords that are assigned to an image from the HTML page containing it will be equally important. We have to determine which words are more important and give them more weights.
- Since many web sites have a common format, images from a particular web site will have many common keywords. We need to reduce the weights of these common words so that images from the same web site are not found to be similar just because they are from the same site.

In this paper, we present the techniques used in AMORE to determine the semantics of images and find semantically similar images. The next section cites related work. Section 3 discusses how AMORE assigns appropriate keywords to images and other media for keyword-based multimedia information retrieval. In Section 4, the method to determine semantically similar images is explained. In Section 5, we describe the evaluation of our schemes showing the effectiveness of our techniques. In Section 6, we introduce AMORE's visual interface with several retrieval scenarios. Various techniques of integrating visual and semantic search are also presented. Finally, Section 7 is the conclusion.

2. Related Work

2.1. WWW Search Engines

There are many popular web search engines like Excite (http://www.excite.com) and Infoseek (http://www.infoseek.com). These engines gather textual information about resources on the web and build up index databases. The indices allow the retrieval of documents containing user-specified keywords. Another method of searching for information on the web is manually generated subject-based directories which provide an useful browsable organization of information. The most popular one is Yahoo (http://www.yahoo.com). However, none of these systems allow image search.

Image search engines for the WWW are also being developed. Excalibur's Image Surfer (http://isurf.yahoo.com) and WebSEEk [3] have built a collection of images that are available on the web. The collection is divided into categories (like automotive, sports, etc.), allowing the users to browse through the categories for relevant images. Moreover, keyword search and searching for images visually similar to a specified image are possible. Alta Vista's Photo Finder (http://image.altavista.com) also allows keyword and visually similar search. However, semantically similar searching is not possible in any of these systems.

WebSeer [4] is a crawler that combines visual routines with textual heuristics to identify and index images of the web. The resulting database is then accessed using a text-based search engine that allows users to describe the image that they want using keywords. The user can also specify whether the required image is a photograph, animation, etc. However, the user cannot specify an image and find similar images.

2.2. Image Searching

Finding visually similar images using image processing techniques is a developed research area. Virage [5] and QBIC [6] are systems for image retrieval based on visual features, which consist of image primitives, such as color, shape, or texture and other domain-specific features. Although they also allow keyword search, the keywords need to be manually specified and there is no concept of semantically similar images.

Systems for retrieving similar images by semantic contents are also being developed [7, 8]. However, in these systems also the semantic content need to be manually associated with each image. We believe that for these techniques to be practical for the WWW, automatic assignment of keywords to the images is essential.

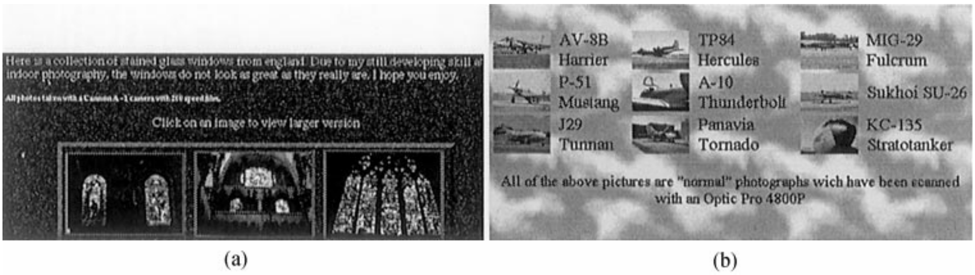
2.3. Assigning Text to WWW Images

Research looking into the general problem of the relationship between images and captions in a large photographic library like a newspaper archive has been undertaken [9, 10]. These systems assume that the captions were already extracted from the pictures, an assumption not applicable to the WWW.

Various techniques have been developed for assigning keywords to images on the WWW. However, none of these techniques can perform reasonably well on all types of HTML pages.

- WebSEEk [3] uses Web URL addresses and HTML tags associated with the images to extract the keywords. This will result in low recall since the surrounding text is not at all considered.
- Harmandas *et al*. [11] use the text after an image URL until the end of a paragraph or until a link to another image is encountered as the caption of the image. They evaluated the effectiveness of retrieval of images based on (a) the caption text, (b) caption text of other images of the page, (c) the non-caption text of the page and (d) the full-text of all pages linked to the image page. However, this method of defining captions will not work with web pages where a collection of images is described by a caption at the top or bottom of all the pages. An example is shown in Figure 1(a). Moreover, indexing an image by the full-text of all pages linked to the image page may result in many irrelevant images being retrieved.
- The Marie-3 system [12] uses text 'near' an image to identify a caption. 'Nearness' is defined as within a fixed number of lines in the parse of the source HTML file. There is an exception if an image occurs within these lines. In this case the caption-scope non-intersection principle is true; it states that the scope for a caption of one image cannot intersect the scope for a caption of another image. Although they found this principle to be true in all their examples, they considered a small section of the web. In some cases, the same caption is used for a collection of images as shown in Figure 1(a). This figure also shows that defining nearness to be a fixed number of lines in the source file will not work because a caption at the top can describe a group of images.
- WebSeer [4] defines the caption of an image to be the text in the same *center* tag as the image, within the same cell of a table as an image or the same paragraph. In our opinion, this system will not assign all the relevant text of an image if it is arranged in a table since it only assigns the text in the same cell as the image. For example, for the table shown in Figure 1(b) the image and the text relevant to it are in different cells.
- The techniques used in commercial systems like Excalibur's and Alta Vista's are not known to the public. However, a preliminary evaluation (by submitting queries and analyzing the retrieved images) indicates that these systems also fail in certain cases; for example if images are formatted in a table.

It should be noted that unlike any of the other image search engines, AMORE allows the user to not only search by keywords but also retrieve semantically similar images.



Here is a collection of stained glass windows from england. Due to my still developing skill at indoor photography, the windows do not look as great as they really are. I hope you enjoy.

All photos taken with a Cannon A-1 camera with 200 speed film.

Click on an image to view larger version

AV-8B Harrier
TP84 Hercules
MIG-29 Fulcrum
P-51 Mustang
A-10 Thunderbolt
Sukhoi SU-26
J29 Tunnan
Panavia Tornado
KC-135 Stratotanker

All of the above pictures are "normal" photographs wich have been scanned with an Optic Pro 4800P

Figure 1. Image caption arrangement schemes where previous approaches may fail: (a) a caption may describe a group of images. (http://fermat.stmarys-ca.edu/∼jpolos/photos/stain.html); (b) images and captions arranged in a table. (http://home1.swipnet.se/∼w-12798/flyg.htm)

3. Assigning Text to WWW Images

In AMORE, during indexing, we use a spider to gather HTML pages from the WWW. These pages are parsed to determine the images contained and referenced from these pages. Besides indexing the images using image processing techniques to find visually similar images during the matching phase, we also need to associate keywords with these images for semantically similar search and retrieving images using keywords. The heuristic algorithm to assign keywords is described in this section.

We believe the images on the web can be classified into two categories: icons and authentic images. Icons are the images whose main function is enhance the 'look' of a web page. They can be substituted by a symbol (e.g. bullets) or by text (e.g. advertising banners), but they are used to make the page more presentable. In contrast to icons, authentic images are the images that cannot be replaced by non-images. We cannot substitute the image of Gogh's painting or the picture of Michael Jordan with text without losing information that we want to deliver. An usability study of AMORE has also shown that people were not interested in the icons when they are using a WWW image retrieval engine. Therefore, these images are not indexed by AMORE and the following discussion is only valid for authentic images of a web page.

3.1. Sources of Keywords

We consider various information available on the web to assign keywords to images. Following are the sources of keywords that we identified useful. We associate all the keywords from these sources to the appropriate image.

- *Image URL*. The URL of an image often describes the image well. For example, for the image http://www.nba.com/finals97/gallery/champions/jordan,floor.jpg, the keywords *nbafinals97*, *jordan*, etc., are very relevant.
- *ALT text*. HTML allows people to annotate an image as an 'ALT = text' attribute. The text associated with the image by ALT is displayed if the image cannot be loaded, and the text also shows up when the mouse cursor stays over the image. Note that although this ALT attribute is the 'official' way to annotate images, most

authors are not aware of it, or simply do not use it; therefore, many images on the web do not have the ALT attribute at all.

- *Headings of the page.* Text appearing in HTML headings in $H\{1–6\}$ tags are also useful. However, only headings before the image are relevant. Moreover, certain headings are ignored by AMORE. For example, consider the following source:

```
⟨BODY⟩
  ⟨H1⟩Top heading⟨/H1⟩
    ⟨H2⟩Section 1 heading⟨/H2⟩
      . . .
      ⟨H3⟩Subsection heading⟨/H3⟩
      . . .
    ⟨H2⟩Section 2 heading⟨/ H2⟩
      . . .
      ⟨IMG SRC = "img.gif"⟩
      . . .
⟨BODY⟩
```

In this case *Section 1 heading* and *subsection heading* are not relevant for the image since they are for a different section. Therefore, whenever we encounter a heading Hi we ignore all text for previous headings Hj if $j \geqslant i$.

- *Page title.* The title of an HTML page gives the general idea behind the page; therefore, it is also useful to understand the semantic content of the image in the page.
- *Anchor text.* When we can get to an image by following a hyperlink, the text in the link describes the linked image well.
- *Text of the embedding page.* In many cases, the text surrounding an image describes the image in detail. Let us now look at the problem of surrounding text assignment.

3.2. Surrounding Text Assignment

Obviously, the text surrounding an image in a web page may be relevant. However, assigning surrounding text to images is a challenging problem; when a web page contains many images, it is ambiguous which part of the text belongs to which image. In this subsection, we will first explain the difficulty of surrounding text assignment, and then we will present our criteria to resolve the ambiguity.

3.2.1. Heterogeneity of Arrangement

Figures 1 and 2 show several examples of how people annotate web images. As we can see from them, there is no predefined or dominant way to arrange images and captions. Sometimes people attach captions before images [Figure 2(a)], sometimes after images [Figure 2(b)] and even sometimes before and after images [Figure 2(c)]. It is also common to use a table to arrange images and captions [Figure 1(b)]. Despite of this heterogeneity, most of the previous works (Section 2.3) assumed that all images in their collection follow a specific arrangement scheme: captions will appear *before* an image or they will appear *after* the image. This assumption may be valid for a carefully selected set

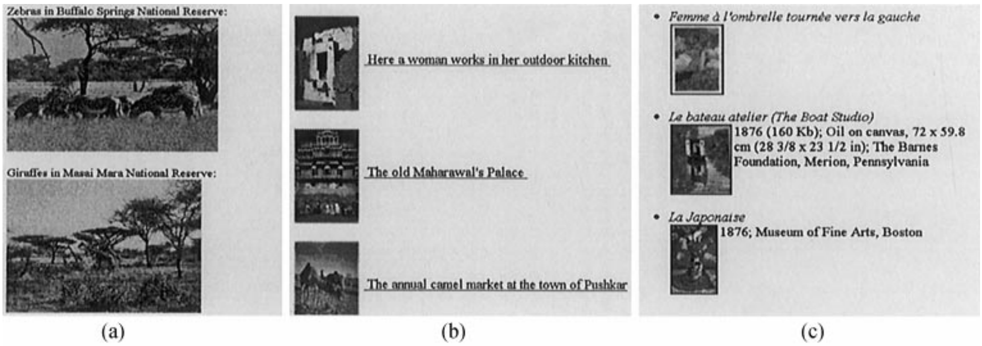

Zebras in Buffalo Springs National Reserve:
Giraffes in Masai Mara National Reserve:
(a)
Here a woman works in her outdoor kitchen
The old Maharawal's Palace
The annual camel market at the town of Pushkar
(b)
• Femme à l'ombrelle tournée vers la gauche
• Le bateau atelier (The Boat Studio)
1876 (160 Kb); Oil on canvas, 72 x 59.8 cm (28 3/8 x 23 1/2 in); The Barnes Foundation, Merion, Pennsylvania
• La Japonaise
1876; Museum of Fine Arts, Boston
(c)

Figure 2. Examples of different kinds of image-caption arrangement schemes: (a) captions before images (http://gaia.ecs.csus.edu/∼reardonf/kenya.html); (b) captions after images (http://photo.net/web-travel/india/wacol); (c) captions both before and after (http://sunsite.unc.edu/wm/paint/auth/cassat)

of collection, but none of them work for a general collection of web pages. Moreover, all previous works defined 'nearness' as the nearness in the source HTML file; therefore, when the final presentation of a page in a browser is different from its arrangement in its source HTML file, previous approaches fail.

3.2.2. *Criteria for Ambiguity Resolution*

Instead of assuming one of the image-caption arrangement schemes, we use the following four criteria to assign text to images.

1. *Visual distance.* The first criterion is as follows: when a text appears in between images, we calculate the visual distance of the text to each image, and we assign the text to the visually closest image.

This first criterion takes advantage of people's general tendency. As we can see from the examples in Figure 2, people usually arrange captions nearby the images that they are describing. Most people feel uncomfortable or aesthetically incorrect, when a caption appears closer to the unrelated image.

More precisely, we formalize the visual distance as follows. We first define a sentence as a sequence of characters without any line breaks or images in the middle. In HTML, line breaks are generated by ⟨BR⟩ or ⟨P⟩ tag, to name a few. We also count the number of line breaks between sentences. For example, ⟨BR⟩ generates one line break and ⟨P⟩ generates two line breaks. We then define a paragraph as a sequence of sentences connected by less than two line breaks. (One line break locates the next sentence right below the previous one, so there is no vertical space between sentences. Therefore, they visually appear as one unit.) The assignment of text to images is done on paragraph level; one whole paragraph is assigned to one image. When it is ambiguous which image a paragraph should be assigned to, i.e. when the paragraph appears in between images, we compare the line break count between the paragraph and the images and assign the paragraph to the closer image. According to this visual distance criterion, all the text in Figure 2 will be assigned to appropriate images.

2. *Syntactic distance.* Sometimes, a text is located at the same visual distance from any of its neighbor images (Figure 3). In these cases, we syntactically compare the text with

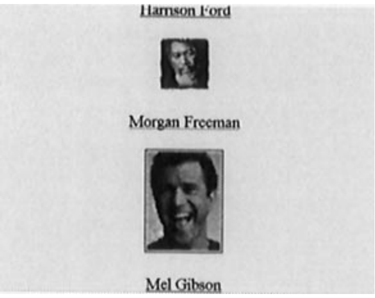


Figure 3. Captions at the same distance from their neighbor images (http://www.pair.com/marilynn/actor.htm)

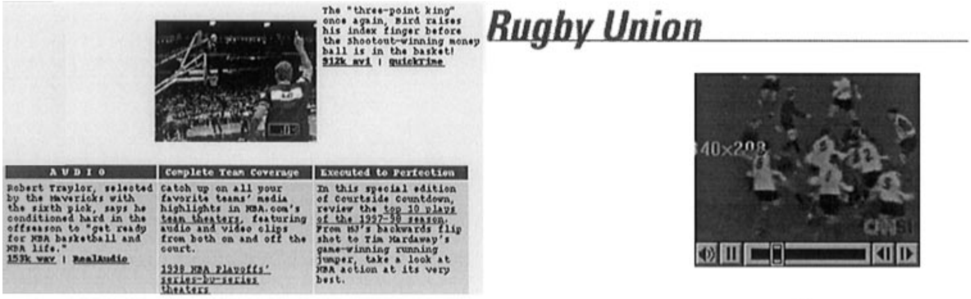
the names of its neighbour images, and we assign the text to the image with the syntactically closest name. The intuition behind this scheme is that people usually give a meaningful name to images. For example, the name of the first image in Figure 3 is ‘freeman’ standing for ‘Morgan Freeman’, and the name of the second image is ‘mgib’ standing for ‘Mel Gibson’. Therefore, the caption ‘Morgan Freeman’ is more similar to the name ‘freeman’ than the name ‘mgib’.

More precisely, we call the file name part of the URL of an image as the *name* of the image. For example, the URL of the first image in Figure 3 is http://www.pair.com/marilynn/freeman.gif, and we name the image as *freeman*. The syntactic distance between the name n and the text t, $d(n, t)$, is defined as $d(n, t) = c(n, t)/|n|$, where $|n|$ is the number of characters in n, and $c(n, t)$ is the number of characters in n that also appear in t in the same order. By this definition, the distance between the name ‘freeman’ and the text ‘Morgan Freeman’ is 1, because all the characters in the name ‘freeman’ also appear in the text in the same order. The distance between the name of the second image, ‘mgib’ and the text ‘Morgan Freeman’ is 0.5, because only m and g appear in the text in the order.

3. *Regular patterns in a table.* In many cases, people follow the same commenting scheme in one page: if a caption appears after an image, it is mostly true for all other captions in the same page. Our third criterion is based on this regularity of image-caption arrangement.

The regular pattern can be easily identified when images and their captions are arranged in a table [Figure 1b]. To detect the regular pattern, we first parse the entire table and scan the table vertically and horizontally. If images appear repeatedly with a fixed interval either vertically or horizontally, then we assign all the text in one interval to the image in that interval. By this criterion we are able to assign the captions in Figure 1(b) to their imges correctly. Note that this example cannot be handled correctly only with visual distance criterion since the captions are at the same distance from their neigbhbor images.

4. *Group of images.* As is shown in Figure 1(a), it is also common that one caption describes a set of following (or preceding) images. We detect these cases by examining whether a sequence of images appear without any text in between. When a set of images appear consecutively, we assign the surrounding text to all images in the set. Note that when a paragraph is located between two sets of images, it is also ambiguous where we



The "three-point king" once again, Bird raises his index finger before the Shootout-winning money ball is in the basket!
512k avi | QuickTime

AUDIO

Robert Traylor, selected by the Mavericks with the sixth pick, says he conditioned hard in the offseason to "get ready for NBA basketball and NBA life."
153k wav | RealAudio

Complete Team Coverage

Catch up on all your favorite teams' media highlights in NBA.com's team theaters, featuring audio and video clips from both on and off the court.

1998 NBA Playoffs' series-by-series theaters

Executed to Perfection

In this special edition of Courtside Countdown, review the top 10 plays of the 1997-98 season. From MJ's backwards flip shot to Tim Hardaway's game-winning running jumper, take a look at NBA action at its very best.

Rugby Union

(a)

(b)

Figure 4. Two ways of specifying multimedia in a HTML file: (a) Anchors (http://www.nba.com/theater); (b) using the *embed* tag. (http://cnnsi.com/rugby/news/1998/11/28/safrica_ireland/rugby.html)

should assign the paragraph. In this case, we apply the visual distance measure to the last (first) images in the preceding (following) set of images and resolve the ambiguity.

3.3. Extension to other Media

There are two ways in which HTML allows the inclusion of other media objects like audio and video.

- The user can specify the multimedia elements as anchors in the HTML page. This is shown in Figure 4(a). Both video and audio can be specified in this way. Thus, in Figure 4(a) videos at the top right and audios at the bottom left are specified as anchors. This is similar to the specifying images as anchors in HTML pages. In this case the anchor text as well as the text surrounding the anchor are useful in determining the semantics of the media.
- The user can also specify video objects with the *embed* tag. This is shown in Figure 4(b). This is similar to specifying images with the *img* tag. Therefore, all the heuristics specified earlier in the section are applicable. However, at present because the number of video available on the WWW is limited, most users do not use tables to organise videos. Therefore, assigning text to video is simpler. It should be noted that since elements like JAVA applets and VRML visualizations are also specified with embed tags, we may be able to guess the semantics of these elements also by our heuristics.

4. Finding Semantically Similar Images

Once appropriate text is assigned to the images using the scheme described in the previous section, these text can be used to retrieve images based on user-specified keywords and also semantically similar images. This section explains the technique to determine semantically similar images. It should be noted that although we emphasize on images, the techniques are also applicable to video and audio contained or referenced in WWW HTML pages.

4.1. Naive Approach

In an earlier version of our system, all the extracted text were represented in a *basic keyword vector model* [13]. It is a well-established research area to find similar objects to a given object. Especially, when the objects are represented as a list of keywords, a substantial number of models have been carefully studied, and the term frequency and inverse document frequency model (*tfidf*) is one of the most popular models.

Under *tfidf* model, to compute similarities between a query object Q and an object P in the collection, we view each object (P or Q) as an m-dimensional vector $W = \langle w_1, \ldots, w_\mathrm{n} \rangle$. The term w_i in this vector represents the ith word in the *vocabulary*, the whole set of keywords. If w_i does not appear in the keyword list of the object, then w_i is zero. If it does appear, w_i is set to represent the significance of the word. The significance of the word w_i is defined as the multiplication of the frequency of, ith word in the object's keyword list to the inverse document frequency (*idf*) of the ith word. The *idf* factor is one divided by the number of times the word appears in the entire 'collection', which in this case would be the concatenation of the keyword lists of all objects. The *idf* factor corresponds to the content discriminating power of a word: a term that appears rarely in documents (e.g., 'Schwarzenegger') has a high *idf*, while a term that occurs in many documents (e.g., 'one') has a low *idf*. Optionally, we may exclude too frequent words thus having no content discriminating power (e.g., 'the') using a *stoplist*. Usually, we normalize the vector W by dividing it by its magnitude $\|W\|$, and the similarity between P and Q can then be defined as the inner product of the normalized P and Q vectors.

However, this technique created problems for determining semantically similar images because obviously, not all keywords are of same importance. Especially, because we consider many keywords from many different sources, a lot of irrelevant keywords are assigned to images. To minimize the problems from irrelevant keywords, we use different weights based on a variety of factors.

4.2. Site Analysis

When all keywords assigned to an image were given equal weights, our system often returned not very related images, just because they resided in the same site as that of the query image. The MOVIEWEB site (http://www.movieweb.com/) illustrates this problem more clearly; all the pages in the MOVIEWEB site have 'MOVIEWEB' in their title and their URL, and they have 'Production Information', 'Release Date', 'Your Comments & Suggestions are Welcome', etc., in common in their body. Therefore, when we did semantic similarity search on an Anthony Hopkins picture from this site, our system returned an image of Batman from this site, before the images of Anthony Hopkins from other sites.

Although the *inverse document frequency* factor sufficiently reduces the weights of the *globally common* keywords, the reduction is not enough for the keywords popular *within a site*; many of them are only *locally* popular, and they are not *globally* popular enough. To reduce the weights of the words popular *within a site*, we also use *inverse in-site frequency*. The inverse in-site frequency reduces the weights of words that occur very often for a particular web site for the images of the web site.

4.3. Sources of Highly Relevant Keywords

Sometimes a long flowing text around an image caused problems. In many cases, when the text around an image is long, it contained many irrelevant keywords and associated them with the image. This long list of irrelevant keywords often deceived our system by matching the irrelevant ones to the keywords of the query image. It is generally difficult to identify *relevant* ones from a list of keywords without understanding the semantic meaning of an image. However, our evaluation showed that following sources frequently give us highly relevant keywords. We give more weights to the keywords from these sources.

- *ALT text and anchor text.* ALT text is the official way to describe an image on the web, and the anchor text is essentially a hint on the linked information. Therefore, people generally give a good explanation of the image in them.
- *Image name.* Very often, the name of an image (the filename part of the URL of an image) gives good hints on its content. We believe this is mainly due to its size limit. The name of an image can be at most 255 characters, so people try to give very compact description of the image in its name. However, the size limit also causes a problem. To save space, people often use abbreviated names (e.g. mj for Michael Jordan), which are not very useful as keywords.

  To identify these abbreviated names we apply *co-occurrence analysis* to image names. When the name is abbreviated, it is generally true that they do not appear in the surrounding text of the image; people do not usually abbreviate words in the main text, and it is unlikely that an unrelated word appears again in the main text. Therefore, we examine whether each word in the image name occurs in the text of the embedding page. If it does not, we do not increase the weight.
- *Page title.* The page title of an image usually gives general description of the image, and it is very useful. However, sometimes the description in the title is too general (e.g., The most fantastic images on the web) or they are not relevant to the content of the image (e.g., JPEG Image 46 kb). We also identify these irrelevant keywords by co-occurrence analysis.
- *Short surrounding text.* Contrary to the *long* flowing text of an image, when the text right after (or before) the image is very *short*, it is generally very relevant to the image. (In our current implementation, a short text is defined as a sentence with less than or equal to 40 characters.) In fact, this is the dominant way people attach a caption to an image on the web.

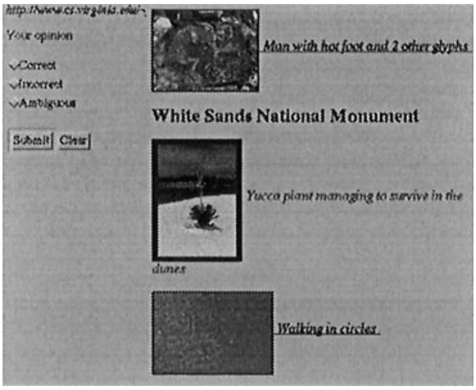
5. Evaluation

5.1. Evaluation of Text Assignment

We evaluated the effectiveness of a text assignment criterion by two measures: *usefulness* and *accuracy*. Usefulness of a criterion measures how many paragraphs were assigned to images by that criterion. If a criterion is used to assign a small number of paragraphs, the criterion may not be worth to adopt. However, this measure alone is not useful because one of the text assignment criteria may have high ‘usefulness’ but may always make an incorrect assignment. Therefore, we also measure the accuracy of a criteria which is

Table 1. The usefulness of text assignment criteria

Images with text resolved by	Number of images	Usefulness
Visual distance	18 804	39%
Syntactic distance	1182	2%
Regular pattern	11 827	24%
Image group	12 458	26%
Total number of images with text	48 297	



Man with hot foot and 2 other glyphs

White Sands National Monument

Yucca plant managing to survive in the dunes

Walking in circles

Figure 5. Interface for measuring the accuracy of a text assignment criterion

determined by the correctness of an assignment of a paragraph to an image by that criteria. Table 1 shows the *usefulness* of our criteria. In this table, the *usefulness* is defined as the percentage of images with a paragraph assigned by the criterion. We can confirm that each criterion assigned a significant number of paragraphs to images. Note that the total number of images in the table is bigger than the sum of other numbers. It is because some web pages had only one image, so we could unambiguously assign the text to the image. Also note that an image may have been assigned many paragraphs, and each of the paragraphs may have been assigned by different criterion.

The second measure, *accuracy*, is defined as C/P, where P is the total number of paragraphs assigned to the images by the criterion, and C is the number of correctly assigned paragraphs. We measured the *accuracy* by human evaluation. We presented the image–paragraph pairs resolved by a criterion to a human evaluator and asked her to decide whether the assignment was correct or not.

Figure 5 shows the evaluation interface. To help the evaluator understand the context of the image, the whole web page containing the image-paragraph pair was shown on the right, with the image in question highlighted and the corresponding paragraph shown in red. After viewing this page, the evaluator decided whether the assignment was *correct*, *incorrect*, or *ambiguous*; *correct* means she believed the assignment was correct, *incorrect* means the paragraph should have been assigned to another image, and *ambiguous* means she could not decide whether it was correct or not.

Table 2. The accuracy of text assignment criteria

Response	Visual	Syntactic	Regular	Group	After[a]	Before[b]
Correct	207	187	221	173	65	150
Incorrect	19	73	22	28	159	59
Ambiguous	48	4	2	51	58	41
Total	274	264	245	252	282	250
Accuracy	92%	72%	91%	86%	23%	72%

[a] Image-*after* text.
[b] Image-*before*-text.

For each criterion, we performed this evaluation for around 300 randomly sampled pairs of paragraphs and images. The images cover a broad range of topics like art, movie, sports, travel, vehicle, and wildlife related images. Although the result of our evaluation may be slightly different from the property of the entire web space, we believe our data set is heterogeneous enough to check the effectiveness of our schemes. In order to compare our criteria with the previous approaches, we generated 300 image–paragraph pairs by assigning paragraphs to the images *before* them and another 300 image–paragraph pairs by assigning paragraphs to the images *after* them. Six human evaluators (members of our research lab) were involved in this evaluation. Note that the evaluators did not know how the image–paragraph association was made. Also note that one image–paragraph pair was evaluated by only one evaluator.

Table 2 shows the result. The accuracy in this table was calculated excluding ambiguous pairs. We can confirm that our visual distance, regular pattern and image group criteria have high accuracy. Especially, the visual distance criterion is much better than the previous approaches. (The regular pattern and the image group criteria are not comparable with the previous approaches.) However, the accuracy difference between the syntactic distance and the image-*before*-text is negligible. Given that the syntactic distance criterion is applied to less than 3% of images, we cannot confirm that our syntactic distance criterion was very helpful. (According to this result, when a paragraph cannot be assigned to an image by the visual distance criterion, we can assume that the paragraph is related to the image before the paragraph, getting 72% accuracy.)

5.2. Evaluation of Finding Semantically Similar Images

The success of finding semantically similar images depends on whether we can identify *relevant* keywords and give them more weight in the vector model. Our technique is based on the sources of the keywords; therefore by examining the *quality* of the keyword sources, we can check whether our heuristic is meaningful. Informally, a keyword source is of *high quality* if it gives only *relevant* keywords. More precisely, the *quality* Q is defined as $Q = R/K$, where K is the total number of keywords from the source and R is the number of relevant keywords. Our method will be successful only when the sources of the highly relevant keywords (those given higher weights) are of high quality.

We measured the quality of the sources by another human evaluation. In this experiment, we presented an image together with a list of keywords and asked the evaluators to select the relevant keywords. Figure 6 shows the interface for this

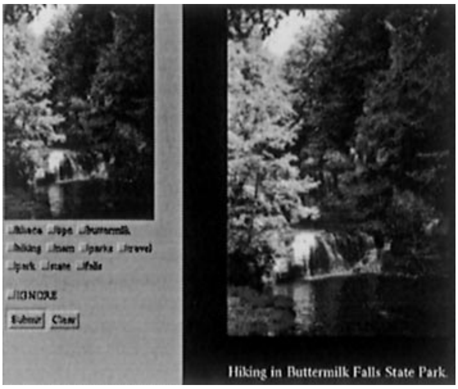


Figure 6. Interface for the evaluation of the quality of the keyword sources

Table 3. The quality of keyword sources

Source	Relevant	Irrelevant	Quality
Image URL	410	781	34%
Image name[a]	201	205	50%
Title[a]	360	222	62%
Alt text[a]	115	18	86%
Anchor text[a]	43	6	87%
Heading	328	275	54%
Surrounding text	901	1767	34%
Short text ($<$ 40 chars)[a]	446	174	72%
Total	1366	2835	33%
Highly relevant section	939	548	63%

[a] Sources of highly relevant keywords.

evaluation. On the right side we presented the web page containing the evaluated image, and on the left side we listed 10 keywords. After reading the page, the evaluator checked the keywords that she considered relevant and left the irrelevant keywords unchecked. To generate the 10 keywords on the left, we first gathered all words from all the sources in Section 3.1, then we eliminated the words in our *stoplist*. Out of the remaining words, we randomly selected 10 words regardless of their source. The evaluation was done for around 400 images by six evaluators. Note that the evaluators did not know where the keywords came from.

Table 3 shows the quality of the sources. To construct this table, we first traced the sources of the keywords examined, and we counted how many relevant/irrelevant keywords each source produced. Note that some keywords came from multiple sources and this is why the total number is smaller than the sum of other numbers. Also note that because of our sampling procedure, the relative numbers in the table correctly represents the relative numbers of keywords from the sources.

From this table, we can confirm that the sources of the highly relevant keywords are the ones with highest quality. Altogether, the quality of our highly relevant keywords is 63% and it covers 69% of relevant keywords.

Table 4. The quality of the sources of highly relevant keywords after co-occurrence and site analysis

Source	Relevant	Irrelevant	Quality
Image name	137	13	91%
Title	275	92	74%
Alt text	115	18	86%
Anchor text	43	6	87%
Short text	424	151	74%
Highly relevant	791	205	79%

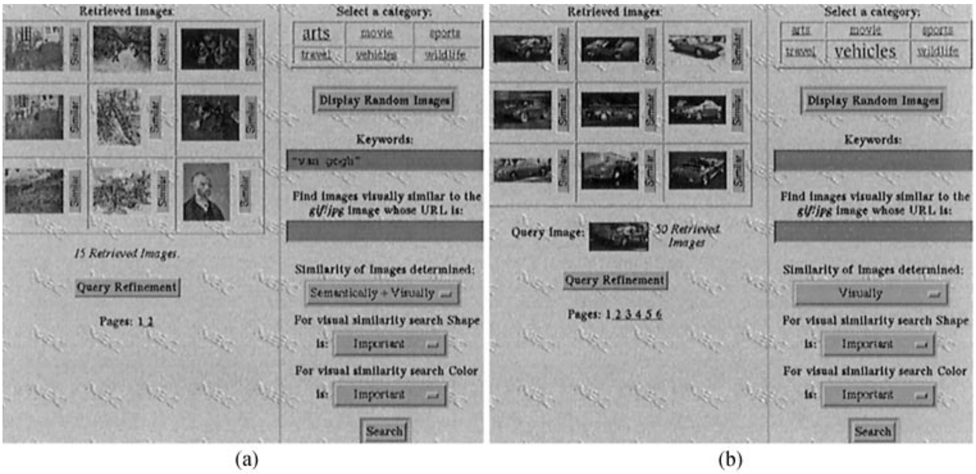
The weights of keywords from highly relevant sources are increased only if they satisfy the co-occurrence criteria (Section 4.3). The weights may also be reduced after site analysis because of the inverse in-site frequency factor (Section 4.2). To measure the effectiveness of the co-occurrence analysis and site analysis, we measured the quality of the keywords with large weights after the analyses, and Table 4 shows the quality. Many of the sources show significant improvement of quality, and the quality of the highly relevant keywords is 79%. Note that although we missed some of the relevant keywords due to the co-occurrence and site analysis, the highly relevant keywords still covers 58% of the relevant keywords.

6. AMORE's Visual Interface

The ideas described in the Sections 3 and 4 have been incorporated in the WWW Image Search engine AMORE. It allows the user to retrieve images of interest using various criteria. For example, the user can search using keywords as shown in Figure 7(a) where the phrase '*van gogh*' was used. The figure presents AMORE's visual query interface which has gone through several rounds of modification based on user feedback. The retrieved images are shown using thumbnails. Like traditional WWW search engines the user can browse through pages of results. We also allow the user to click on the *Similar* button next to a thumbnail and retrieve similar images. This visual navigation strategy is helpful in quickly retrieving the target images.

For the similarity search, the user can specify whether the similarity is visual or semantic. For example, Figure 7(b) shows the images retrieved by a visual similarity search when clicking on a red *Acura NSX* car. Images of other red cars are retrieved although they are not Acuras. On the other hand, for semantic similarity search, images of other *Acura NSX* cars are retrieved even though they are not visually similar as shown in Figure 8(a).

Sometimes, the user may be confused why an image was retrieved by semantic similarity search. For example, just by looking at the images of Figure 8(a) it is not obvious why the retrieved images are similar to the query image. Our initial user studies show that explaining to a naive user why an image was retrieved is essential. Therefore, when the user clicks on a retrieved image, AMORE provides such feedback. This is shown in Figure 8(b). All the keywords assigned to an image from different sections of the page (Title, Heading, Caption, etc.) are shown. The matching keywords with the



(a) Keywords: "van gogh"

15 Retrieved Images. Pages: 1 2

Find images visually similar to the gif/jpg image whose URL is:

Similarity of Images determined: Semantically + Visually

For visual similarity search Shape is: Important

For visual similarity search Color is: Important

(b) Query Image: 50 Retrieved Images

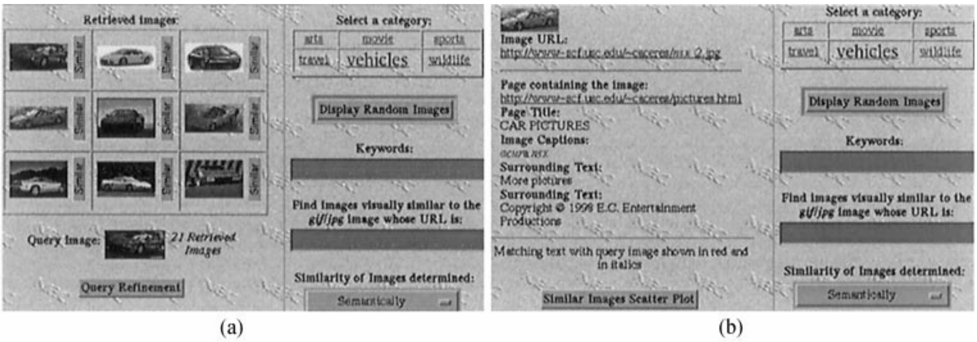
Pages: 1 2 3 4 5 6

Similarity of Images determined: Visually

For visual similarity search Shape is: Important

For visual similarity search Color is: Important

Figure 7. Examples of different kind of AMORE searches: (a) keyword search with "*Van Gogh*"; (b) visual similarity search with a red *Acura NSX* car



(a) Query Image: 21 Retrieved Images

Find images visually similar to the gif/jpg image whose URL is:

Similarity of Images determined: Semantically

(b) Image URL:
http://www-scf.usc.edu/~caceres/nsx 2.jpg

Page containing the image:
http://www-scf.usc.edu/~caceres/pictures.html

Page Title:
CAR PICTURES

Image Captions:
acura nsx

Surrounding Text:
More pictures

Surrounding Text:
Copyright © 1998 E.C. Entertainment Productions

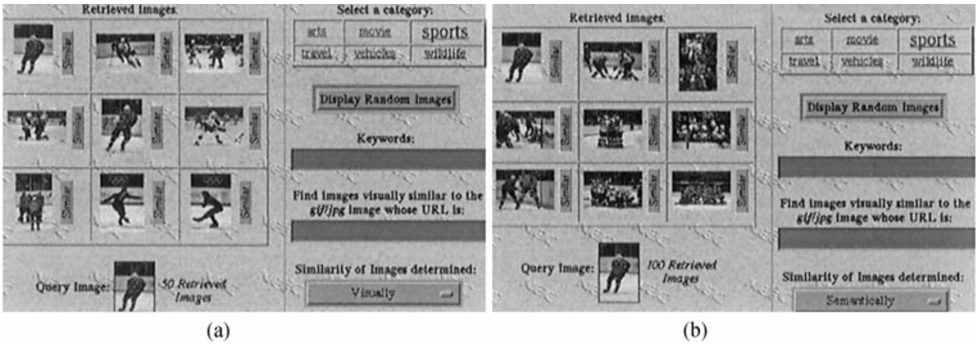
Matching text with query image shown in red and in italics

Figure 8. Semantic similarity search and the feedback provided to the user: (a) semantic similarity search with a red *Acura NSX* car; (b) details of a retrieved image

query image are shown in red and italics. Thus, Figure 8(b) explains that the image was retrieved because it was an *Acura NSX*. Note that we give different weights to a word depending on the section of the page from where it retrieved as explained in Section 4. However, to avoid confusion, this is not shown to the user.

6.1. Integrating Visually and Semantically Similar Search

Because the visual similarity does not consider the semantics of the images, sometimes it may retrieve images not meaningful to the user. For example, Figure 9(a) shows the images retrieved by a visual similarity search when clicking on the photo of an ice hockey player. Although photos of other ice hockey events are retrieved, we also retrieve photos of figure skating (the last two images). These images are visually similar, but they may



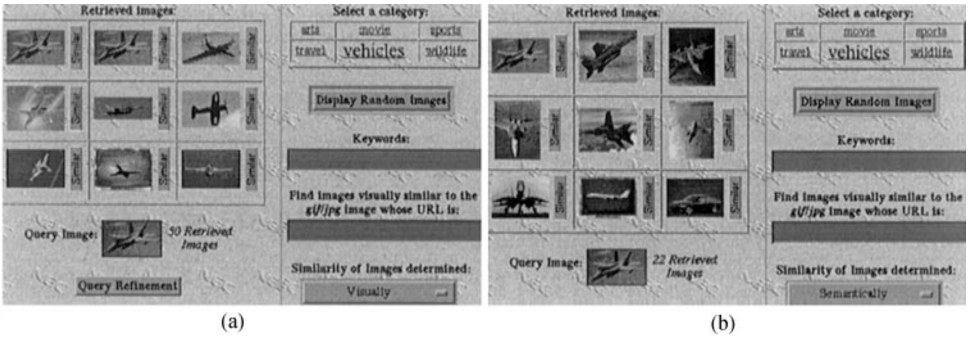
Query Image: 50 Retrieved Images

Similarity of Images determined: Visually

Query Image: 100 Retrieved Images

Similarity of Images determined: Semantically

Figure 9. Example where semantic similarity works better: (a) visual similarity search with an Ice Hockey player; (b) semantic similarity search with an Ice Hockey player



Query Image: 50 Retrieved Images

Similarity of Images determined: Visually

Query Image: 22 Retrieved Images

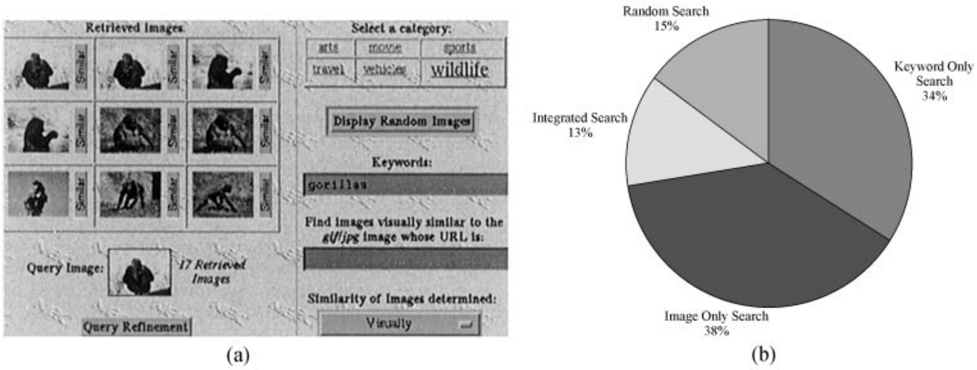
Similarity of Images determined: Semantically

Figure 10. Example where visual similarity works better: (a) visual similarity search with a plane picture; (b) semantic similarity search with a plane picture

not be meaningful. Semantic similarity search, on the other hand, retrieves only ice hockey images as shown in Figure 9(b).

However, the success of semantic similarity search depends on relevant keywords being assigned to the images. Although our techniques remove irrelevant keywords in most cases, sometimes such keywords cause irrelevant images to be retrieved. An example is shown in Figure 10. The query image belongs to a page whose title was *Flight and Car Pictures*. Therefore, a car picture is also retrieved [the last picture of Figure 10 (b)]. In this case, visual similarity worked better [Figure 10(a)].

A solution to the problem of irrelevant images being retrieved by visual or semantic search alone is to integrate similarity search with keyword search. An example is shown in Figure 11(a). Visual similarity search of a picture of gorilla is integrated with the keyword *gorillas* to retrieve very relevant images. Although this kind of search provides the best result, it is not as popular as we would have linked. Figure 11(b) shows the percentage of the different kind of search that actual web users of AMORE used. This information was obtained by an analysis of AMORE's access logs. (More than 50 000 accesses were examined.) It shows that keyword only and visual similarity search were the most common. (At the time of the analysis semantic similarity search was not

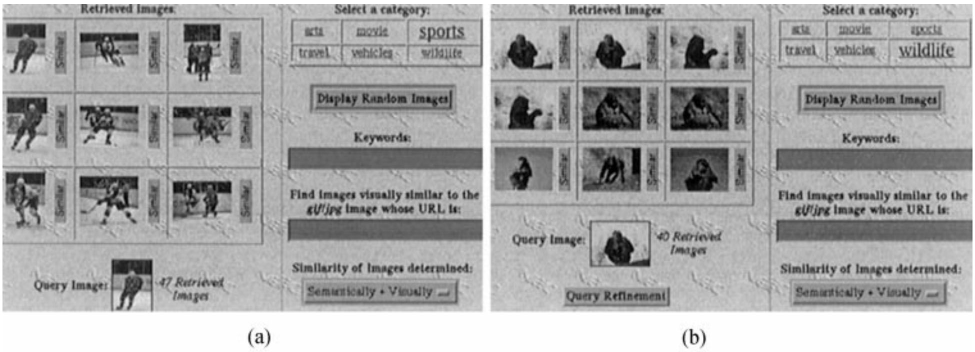


Keywords: gorillas

Find images visually similar to the gif/jpg image whose URL is:

Similarity of Images determined: Visually

Figure 11. Although integrated keyword and similarity search works best, it is not very popular: (a) integrating visual similarity and keyword search; (b) comparing the popularity of the different searching techniques



Find images visually similar to the gif/jpg image whose URL is:

Similarity of Images determined: Semantically + Visually

Figure 12. Integrated semantic and visual similarity search: (a) Ice Hockey image; (b) gorilla image

available). Integrated search was even less popular than random search (in which some random images were shown).

We believe that integrating visual and semantic similarity search will be very useful for the users of AMORE. This can be done in various ways. Let us now look at three different methods of integration.

6.1.1. Semantic Similarity followed by Visual Similarity

A straightforward approach is to first retrieve images by semantic similarity and apply visual similarity to only the retrieved images. This generally retrieves visually similar images that are also semantically meaningful. Thus, in Figure 12(a), we retrieved ice hockey pictures both visually and semantically similar to the query image (cf. with Figure 9). Similarly, the images shown in Figure 12(b) are very relevant. It is interesting that the retrieved images in this case are almost identical to the integrated keyword and visual similarity search shown in Figure 11(a). We are planning to make this kind of integrated search the default similarity search technique in AMORE. Obviously, visual

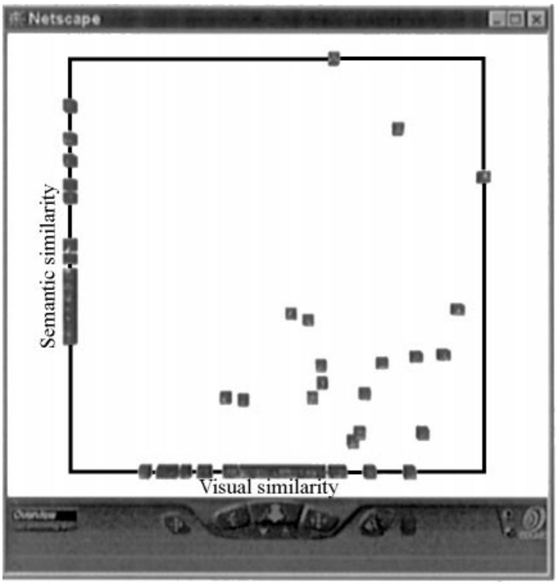


Figure 13. A scatterplot visualization of the results of the image search. The *X* and *Y* dimensions are mapped to visual and semantic similarities, respectively

and semantic similarity searching alone are also useful in some cases as shown in Figures 7(b) and 8(a). Therefore, we allow the user to change to other techniques of similarity search using the Forms interface shown in Figure 12.

6.1.2. Scatterplot Visualization

The HTML interface allows the user to only rank the retrieved images either visually or semantically. It will be also useful for the user to see how much the retrieved images are visually and semantically similar in one screen. A scatterplot visualization can be used for this purpose.

Figure 13 shows the scatterplot visualization developed in VRML. The visualization is generated dynamically when the user clicked the *Similar Images Scatter Plot* button in the image information page [Figure 8(b)]. For the top 50 semantically and visually similar images, both the semantic and visual similarity values are calculated. (Note that some images may be both visually and semantically similar.) The images are represented by cubes with the images shown as texture maps. The visual and semantic similarity values are mapped onto the *X* and *Y* dimensions, respectively. (The user-specified image obviously has a high value of *x* and *y* and is not shown). The visualization gives a good idea of how the other images match the query image. An interesting observation is that most images are either visually similar (high values for visual similarity and thus large *x* values) or semantically similar (large *y* values). Only few images are both visually and semantically similar. This kind of observation is not easily apparent if the results were shown in the traditional HTML interfaces.

The user can navigate through the space to any image of interest using the facilities of the VRML browser (SGI's Cosmo Player is used in Figure 13). Clicking on the image

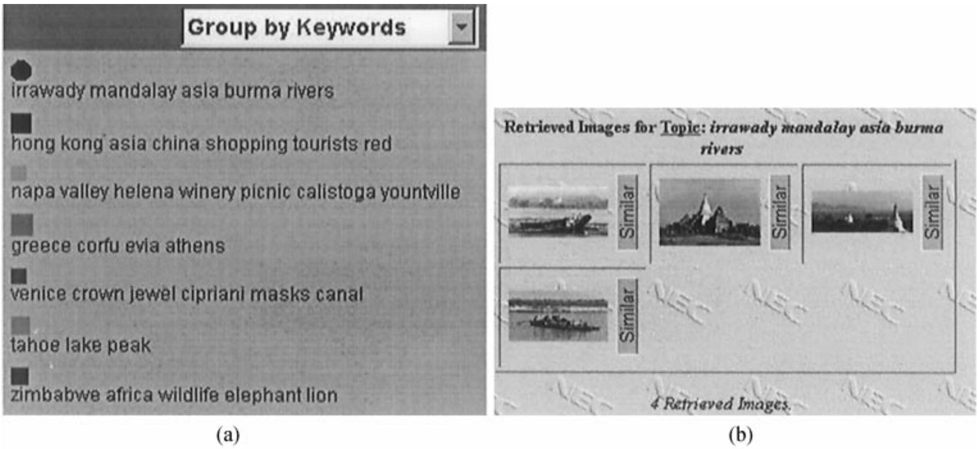

Group by Keywords
irrawady mandalay asia burma rivers
hong kong asia china shopping tourists red
napa valley helena winery picnic calistoga yountville
greece corfu evia athens
venice crown jewel cipriani masks canal
tahoe lake peak
zimbabwe africa wildlife elephant lion
Retrieved Images for Topic: irrawady mandalay asia burma rivers
Similar
Similar
Similar
Similar
4 Retrieved Images.
(a)
(b)

Figure 14. Grouping by keywords the results of an image search: (a) query result visualization environment applet; (b) retrieved images of cluster for Burma

retrieves more information about the image. Different viewpoints are also provided that allows the user to zoom into the image that is most semantically or visually similar.

6.1.3. Organizing the Results of Visual Similarity Search by Text Clustering

Like text search engines, AMORE sometimes retrieves a lot of images. The results are shown in several pages, nine images a page. This is obviously not very user-friendly. Therefore, we have developed a *Query Result Visualization Environment* (*QRVE*). This is a JAVA applet (available from the *Query Refinement* button) that allows the user to organize the results in various ways. For example, the user can organize the retrieved images by the web sites from which they were retrieved.

Another method of organizing the results is by text clustering. This is shown in Figure 14. Figure 14(a) shows the JAVA applet in which the retrieved images are grouped into clusters based on the keywords relevant to them. Each cluster if represented by a graphical element (*glyph*). The main topics of the cluster are shown at the bottom of the glyph. From the cluster topics the user gets an idea of the types of images that will be present in the cluster. For example, the cluster with topics *burma*, *irrawady*, etc., will have pictures of Burma (Irrawady is a river of Burma) as shown in Figure 14(b). Besides, issuing another search with a retrieved image from one of the clusters, the user can also refine the search by looking at all the images of a cluster. Since the clusters are formed by text analysis, by refining, semantic and visual search are smoothly integrated; even if the user does not combine semantic and visual search during the querying phase, they are combined in the browsing phase.

Note that the glyph is a circle for the currently selected cluster and square for the others. The size of the glyph is proportional to the number of images retrieved from that group. Different shades of grey are used to represent how similar the images of a cluster are, on average, to the query image.

We use a graph-theoretic method to form clusters. We consider each image to be a node in the graph. A link exists between two nodes in the graph if the similarity between the two corresponding images is greater than a threshold. Initially the threshold is 0.1. We find the *connected components* in the graph. The set of nodes in a connected component form a cluster. If a cluster has too many nodes, the threshold is increased and the process repeated for the nodes of the cluster. This is the *single-link* cluster formation technique which has been extensively used in information retrieval and found to be one of the most appropriate [14]. Note that in this case each image can belong to only one cluster or none at all (in which case they are grouped in a cluster named *Others*).

The clusters are represented in the Java applet by the main topics of the clusters. The topics of a cluster are the words that appear in most of the images of the cluster. Since adverbs, prepositions, etc., are not very interesting they are not allowed to be cluster topics. (We use Wordnet [15] to determine the figure of speech.) Moreover, if a word appears as a topic in many clusters, it is removed as a topic word.

7. Conclusion

Finding the target image from a large image database is a difficult problem. Since an image can be described by various characteristics, an effective image search engine should provide the users various techniques to reach the target. We have developed the WWW image search engine AMORE which allows the user three mechanisms to find the images of interest. The user may specify keywords and also find images visually or semantically similar. Moreover, visual and semantic search can be integrated.

In this paper, we discussed the criteria we used to assign keywords to WWW images and the technique to retrieve semantically similar images. We use unique criteria like using visual and syntactic distance between images and potential captions to resolve ambiguities while assigning text to images. Techniques like co-occurrence analysis and site analysis are used to determine relevant and less relevant keywords. This information is used to retrieve semantically similar images using a vector space model. Our techniques can be applied to other multimedia elements found on the WWW.

The effectiveness of a information-retrieval engine is also dependent on the user interface to specify the queries and show the retrieved images. We believe that AMORE's visual interface is simple and easy to use. It allows various ways to integrate visual and semantic search. The retrieval scenarios presented in the paper show the effectiveness and the usefulness of AMORE.

Future work is planned along various directions:

- *Global evaluation*: one limitation of our evaluation described in Section 5 is that although it shows that the individual methods were effective, it does not ensure the global effectiveness of AMORE. Therefore, an evaluation of the AMORE system based on the standard IR matrices of *precision* and *recall* is essential. We will need to tune the weights given to the different sources of keyword based on the evaluation.
- *Usability study*: we are exploring other techniques of integrating visual and semantically similar search. We are also planning to perform an usability study of the user interfaces of integrated search. This will help us to determine the best method of integration.

Our ultimate objective is to develop a multimedia information retrieval engine that allows the user to retrieve interesting media from the WWW using various easy-to-use techniques.

References

1. S. Mukherjea, K. Hirata & Y. Hara (1997) Towards a multimedia world-wide web information retrieval engine. In: *Proceedings of the 6th International World-Wide Web Conference*, Santa Clara, CA, April, pp. 177–188.
2. K. Hirata, Y. Hara, N. Shibata & F. Hirabayashi (1993) Media-based Navigation for Hypermedia Systems. In: *Proceedings of ACM Hypertext '93 Conference*, Seattle, WA, November, pp. 159–173.
3. S. Chang, J. Smith, M. Beigi & A. Benitez (1997) Visual information retrieval from large distributed online repositories. *Communications of ACM* 40, 63–71.
4. C. Frankel, M. Swain & V. Athitsos (1996) WebSeer: an image search engine for the world-wide web. Technical Report 94-14, Computer Science Department, University of Chicago, August.
5. J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Jain & C. Shu (1996) The virage image search engine: an open framework for image management. In: *Proceedings of the SPIE—The International Society for Optical Engineering: Storage and Retrieval for Still Image and Video Database IV*, San Jose, CA, U.S.A., February.
6. M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele & P. Yanker (1995) Query by image and video content: the QBIC system. *IEEE Computer* 28, 23–48.
7. A. Smeaton & I. Qigley (1996) Experiments on using semantic distances between words in image caption retrieval. In: *Proceedings of the ACM SIGIR '96 Conference on Research and Development in Information Retrieval*, Zurich, Switzerland, August, pp. 174–180.
8. Y. Aslandogan, C. Thier, C. Yu, J. Zou & N. Rishe (1997) Using semantic contents and WordNet in image retrieval. In: *Proceedings of the ACM SIGIR '97 Conference on Research and Development in Information Retrieval*, Philadelphia, PA, July, pp. 286–295.
9. R. Srihari (1995) Automatic indexing and content-based retrieval of captioned images. *IEEE Computer* 28, 49–56.
10. N. Rowe (1996) Using local optimality criteria for efficient information retrieval with redundant information filters. *ACM Transactions on Information Systems* 14, 138–174.
11. V. Harmandas, M. Sanderson & M. Dunlop (1997) Image retrieval by hypertext links. In: *Proceedings of the ACM SIGIR '97 Conference on Research and Development in Information Retrieval,* Philadelphia, PA, July, pp. 296–303.
12. N. Rowe & B. Frew (1998) Automatic caption localization for photographs on World-Wide Web pages. *Information Processing and Management* 34, 95–107.
13. G. Salton & M. McGill (1983) *Introduction to Modern Information Retrieval.* McGraw-Hill, New York.
14. C. Van-Rijsbergen (1979) *Information Retrieval.* Butterworths, London.
15. G. Miller (1995) WordNet: a lexical database for English. *Communications of the ACM* 38, 39–41.