An Image and Video Search Engine for the World-Wide Web

John R. Smith and Shih-Fu Chang

Department of Electrical Engineering and
Center for Image Technology for New Media
Columbia University,
New York, N.Y. 10027

{jrsmith, sfchang}@itnm.columbia.edu

Abstract

We describe a visual information system prototype for searching for images and videos on the World-Wide Web. New visual information in the form of images, graphics, animations and videos is being published on the Web at an incredible rate. However, cataloging this visual data is beyond the capabilities of current text-based Web search engines. In this paper, we describe a complete system by which visual information on the Web is (1) collected by automated agents, (2) processed in both text and visual feature domains, (3) catalogued and (4) indexed for fast search and retrieval. We introduce an image and video search engine which utilizes both text-based navigation and content-based technology for searching visually through the catalogued images and videos. Finally, we provide an initial evaluation based upon the cataloging of over one half million images and videos collected from the Web.

Keywords – content-based visual query, image and video storage and retrieval, World-Wide Web.

1 Introduction

A large number of catalogs and search engines index the plethora of documents on the World-Wide Web. For example, recent systems, such as Lycos, Alta Vista and Yahoo, index the documents by their textual content. These systems periodically scour the Web, record the text on each page and through processes of automated analysis and/or (semi-) automated classification, condense the Web into compact and searchable indexes. The user, by entering query terms and/or by selecting subjects, uses these search engines to more easily find the desired Web documents. Generally, the text-based Web search engines are evaluated on the basis of the size of the catalog, speed and effectiveness of search, and ease of use [1].

However, no tools are currently available for searching for images and videos. This absence is particularly notable given the highly visual and graphical nature of the Web [2]. Visual information is published both as embedded in Web documents and as stand-alone objects. The visual information takes the form of images, graphics, bitmaps, animations and videos. As with Web documents in general, the publication of visual information is highly volatile. New images and videos are added everyday and others are replaced or removed entirely. In order to catalog the visual information, a highly efficient automated system is needed that regularly traverses the Web, detects visual information and processes it in such a way to allow for efficient and effective search and retrieval.

We recently developed an image and video search engine[†] to fulfill this need [3]. The system collects images and videos from the Web and provides tools for searching and browsing through the collection.

[†] `http://www.itnm.columbia.edu/webseek`

The system is novel in that it utilizes text and visual information synergically to provide for cataloging and searching for the images and videos. The complete system possesses several powerful functionalities, namely, (1) searching using content-based techniques, (2) query modification using content-based relevance feedback, (3) automated collection of visual information, (4) compact presentation of images and videos for displaying query results, (5) image and video subject search and navigation, (6) text-based searching, and (7) search results manipulations such as intersection, subtraction and concatenation.

1.1 Outline

In this paper, we describe the complete system for cataloging and searching for images and videos on the Web. In section 2, we describe the process for automated collection of the visual information. In section 3, we describe the procedures for classifying the collected images and videos using key-term mappings and directory names. We also present and utilize a new taxonomy for visual information. In section 4, we describe the system for navigating through subject classes, searching, viewing query results and manipulating the search results lists. In section 5, we describe several content-based tools for searching, browsing and revising queries. In particular, we describe the system's utilization of color histograms for the content-based manipulation of images and videos. Finally, in section 6, we provide an initial evaluation of the system in the collection of more than one half million images and videos from 16,773 sites on the World-Wide Web.

2 Image and Video Collection Process

The image and video collection process is conducted by an autonomous Web agent or spider. The agent traverses the Web by following the hyperlinks between documents. It detects images and videos, retrieves and processes them and adds the new information to the catalog. The overall collection process, illustrated in Figure 1, is carried out using several distinct modules: (1) the *Traversal Spider* – assembles lists of candidate Web pages that may include images, videos or hyperlinks to them, (2) the *Hyperlink Parser* – extracts the *URL*s of the images and videos, and (3) the *Content Spider* – retrieves and analyzes the images and videos.

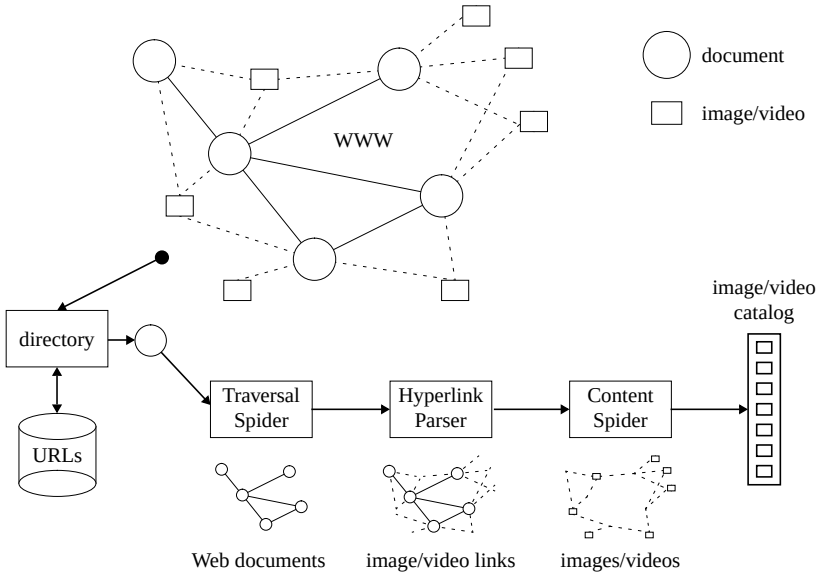


Figure 1: Image and video gathering process.

2.1 Image and Video Detection

In the first phase, the *Traversal Spider* traverses the Web looking for images and videos, as illustrated in Figure 2. Starting from seed *URL*s, the *Traversal Spider* follows a breadth-first search across the Web. It retrieves pages via Hypertext Transfer Protocol (*HTTP*) and passes the Hypertext Markup Language (*HTML*) code to the *Hyperlink Parser*. In turn, the *Hyperlink Parser* detects new *URL*s, encoded as *HTML* hyperlinks, and adds them back to the queue of Web pages to be retrieved by the *Traversal Spider*. In this sense, the *Traversal Spider* is similar to many of the conventional spiders or robots that follow hyperlinks in some fashion across the Web [4]. The *Hyperlink Parser* detects the hyperlinks in the Web documents and converts the relative *URL*s to absolute addresses. By examining the types of the hyperlinks and the filename extensions of the *URL*s, the *Hyperlink Parser* assigns each *URL* to one of several categories: image, video or *HTML*. The mapping between filename extensions and Web object type is given by the *Multipurpose Internet Mail Extensions* (MIME) content type labels.

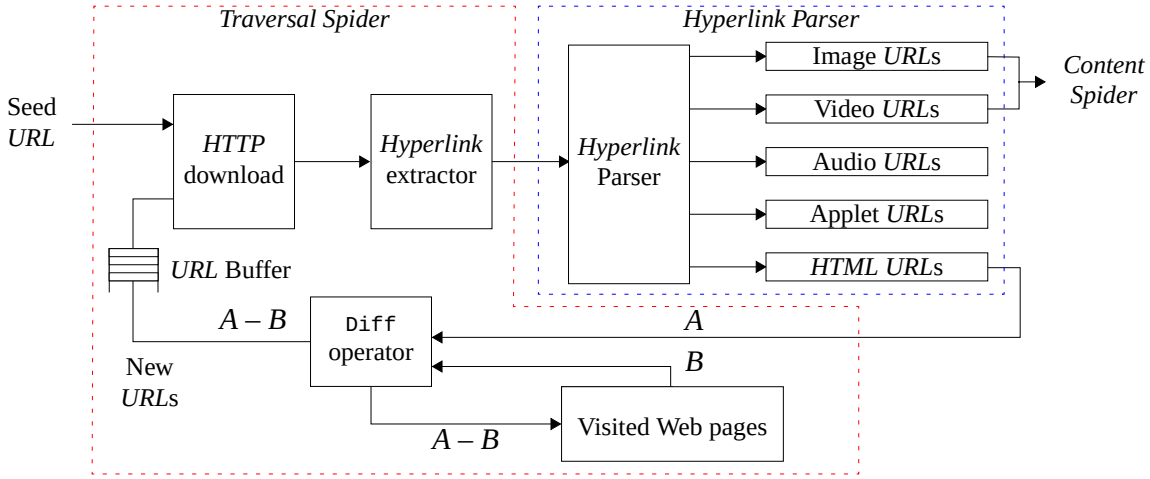


Figure 2: The *Traversal Spider* traverses the Web and assembles the list of *URL*s of images and videos.

In the second phase, the list of image and video *URL*s from the *Hyperlink Parser* is passed to the *Content Spider*. The *Content Spider* retrieves the images and videos, processes them and adds them to the catalog. Three important functions of the *Content Spider* are to (1) extract visual features that allow for content-based techniques in searching, browsing and grouping, (2) extract other attributes such as width, height, number of frames, type of visual data, and so forth, and (3) generate an icon, or motion icon, that sufficiently compacts and represents the visual information. The process of extracting visual features from the images and videos generates color histograms, which are discussed in Section 5. The other attributes of the images and videos populate the database tables, which are defined in Section 3.4. Finally, the *Content Spider* generates coarse and highly compressed versions of the images and videos to provide pictorial data in the query output.

2.1.1 Image and Video Presentation

For images, the coarse versions are obtained by reducing and compressing the originals where the compression format, either JPEG or GIF, is chosen to match the original image format. For video, the coarse versions are generated by subsampling the original video both spatially and temporally. The temporal subsampling is achieved in a two step process: first, one frame is kept per every one second of video. Next, scene change detection is performed on the frames to detect the key frames of the sequence [5]. This allows

for the elimination of duplicate scenes in the coarse version. Finally, the video is re-animated from the key frames and packaged as an animated GIF file. In the query results the coarse videos appear to the user as animated samples of the original video.

3 Subject Classification and Indexing

Utilization of text is essential to the cataloging process. In particular, every image and video on the Web has a unique Web address and possibly other *HTML* tags, which provide for valuable interpretation of the visual information. We process the Web addresses, or *URL*s, and *HTML* tags in the following ways to index the images and videos: extract **terms**, extract **directory names**, map **key-terms** to **subjects** using a **key-term dictionary**, and map **directory names** to **subjects**.

3.1 Text Processing

Images and videos are published on the Web in two forms: *inlined* and *referenced*. The *HTML* syntax differs in the two cases. To inline, or embed, an image or video in a Web document, the following code is included in the document: `<img src=URL alt=`[alt text]`>`, where `URL` gives the relative or absolute address of the image or video. The optional `alt` tag specifies the text that appears when the browser is loading the image/video. Alternatively, images and videos may be referenced from parent Web pages using the following code: `<a href=URL>`[hyperlink text]`</a>`, where the optional [hyperlink text] provides the high-lighted text that describes the image/video pointed to by the hyperlink.

3.1.1 Term Extraction

The **terms**, t_k's, are extracted from the image and video *URL*s, *alt* tags and hyperlink text by chopping the text at non-alpha characters. For example, the *URL* of an image or video has the following form

`URL = http://host.site.domain[:port]/[ user/][directory/][file[.extension]].`

Here `[...]` denotes an optional argument. For example, several typical *URL*s are

URL_1 = `http://www.mynet.net:80/animals/domestic-beasts/dog37.jpg`,
URL_2 = `http://camille.gsfc.nasa.gov/rsd/movies2/Shuttle.gif`,
URL_3 = `http://www.arch.columbia.edu/DDL/projects/amiens/slides/slide6b.gif`.

Terms are extracted from the `directory` and `file` strings using functions $\mathcal{F}_{\text{key}}$ and $\mathcal{F}_{\text{chop}}$ where

$$\mathcal{F}_{\text{key}}(\text{URL}) = \mathcal{F}_{\text{chop}}(\texttt{directory/file}),$$

and $\mathcal{F}_{\text{chop}}(\text{string})$ gives the set of substrings that are delimited by non-alpha characters. For example,

$$\begin{aligned} \mathcal{F}_{\text{key}}(\text{URL}_1) &= \mathcal{F}_{\text{chop}}(\text{“animals/domestic-beasts1/dog37”}) \\ &= \text{“animals”, “domestic”, “beasts”, “dog”.} \end{aligned}$$

The process of extracting terms produces an overall set $\{t_k\}$ of terms for the image and video collection. The system indexes the images and videos directly using inverted files [6] on the term set $\{t_k\}$. In addition, certain terms, key-terms, t_k^*'s, are used to map the images and videos to subject classes, as we explain shortly.

3.1.2 Directory Name Extraction

A **directory name**, d_l, is a phrase extracted from the *URL*s that groups images and videos by location on the Web. The directory name consists of the directory portion of the *URL*, namely, $\mathcal{F}_{\text{dir}}(\text{URL}) =$ `directory`. For example, $\mathcal{F}_{\text{dir}}(\text{URL}_1) =$ "animals/domestic-beasts". The process of extracting directory names produces an overall set $\{d_l\}$ of directory names for the image and video collection. The directory names are used by the system to map images and videos to subject classes.

3.2 Image and Video Subject Taxonomy

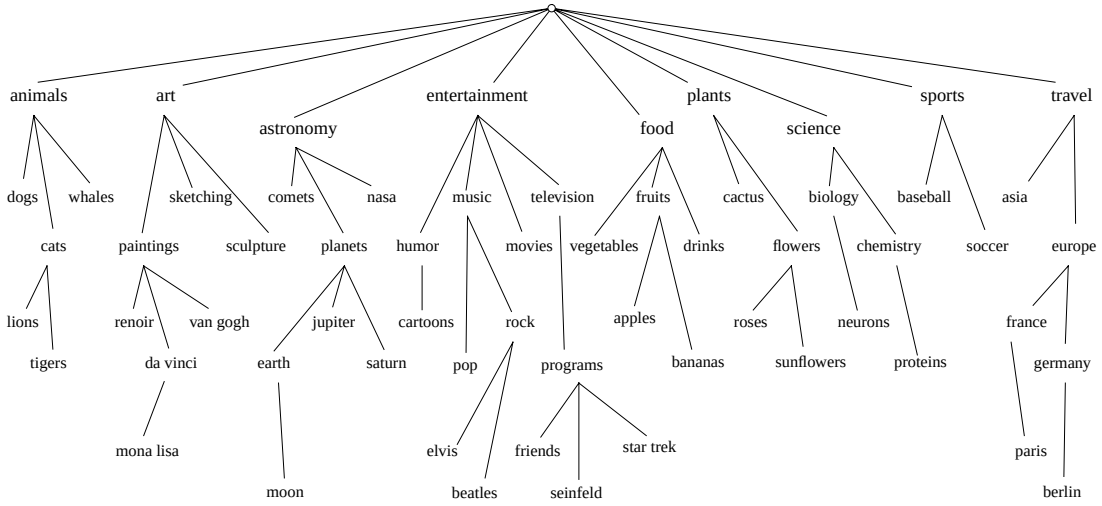


Figure 3: Portion of the image and video subject taxonomy $\{\widehat{s_m}\}$.

A **subject class** or **subject**, s_m, is an ontological concept that represents the semantic content of an image or video, i.e., subject = "basketball". A **subject taxonomy** is an arrangement of the set of subject classes $\{s_m\}$ into a hierarchy, denoted as $\{\widehat{s_m}\}$. In the process of inspecting the terms for key-term mappings, we are developing the subject taxonomy for image and video topics. A portion is illustrated in Figure 3. For example, when a new and descriptive term, such as $t_k =$"basketball" is detected and added to the key-term dictionary, we add a corresponding subject class to the taxonomy if it does not already exist, i.e., $s_m =$"sports/basketball".

3.3 Key-term Dictionary and Directory Name to Subject Mappings

The **key-terms**, t_k^*'s, are terms that are semantically related to one or more subject classes, s_m's. The **key-term dictionary** contains the set of key-terms $\{t_k^*\}$ and related subject classes s_m. As such, the key-term dictionary provides a set of mappings $\{\mathcal{M}_{km}\}$ from key-terms to subject classes, where

$$\mathcal{M}_{km} \ : \ t_k^* \rightarrow s_m . \tag{1}$$

We build the key-term dictionary in a semi-automated process. In the first stage, the term histogram for the image and video archive is computed, such that each term t_k is assigned a count number f_k which indicates how many times the term appeared. Then the terms are ranked by highest count f_k and are presented in this order of priority for manual assessment.

The goal of the manual assessment is assign qualified terms $t_k^* \in \{t_k\}$ to the key-term dictionary. The qualified terms should be descriptive and not homonymic. Ambiguous terms make poor key-terms. For

example, the term "rock" is a not a good key-term due to its ambiguity. "Rock" may refer to a large mass of stone, rock music, a diamond, or several other things. Once a term and its mappings are added to the key-term dictionary, it applies to all images and videos.

term: t_k	count: f_k
image	86380
gif	28580
icon	14798
pic	14035
img	14011
graphic	10320
picture	10026
small	9442
art	8577

(a)

key-term: t_k^*	count: f_k	mapping $\mathcal{M}_{km}$ to subject s_m
planet	1175	astronomy/planets
music	922	entertainment/music
aircraft	458	transportation/aircraft
travel	344	travel
gorilla	273	animals/gorillas
starwars	204	entertainment/movies/films/starwars
soccer	195	sports/soccer
dinosaur	180	animals/dinosaurs
porsche	139	transportation/automobiles/porsches

(b)

Table 1: Sample (a) terms and their counts $\{t_k : f_k\}$ and (b) key-terms counts $\{t_k^* : f_k\}$ with subjects s_m's and mappings $\mathcal{M}_{km} : t_k^* \rightarrow s_m$'s. Taken from the assessment of over $500,000$ images and videos.

From the initial experiments of cataloging over $500,000$ images and videos, the terms listed in Table 1 are a sample of those extracted. Notice in Table 1(a) that some of the most frequent terms are not sufficiently descriptive of the visual information, i.e., terms "image", "picture". However, the terms in Table 1(b) unambiguously indicate the subject of the images and videos, i.e., terms "aircraft", "gorilla", "porsche". These key-terms are used to classify the images and videos into subject classes. For example, we added the key-terms t_k^*'s and corresponding subject mappings $\mathcal{M}_{km}$'s illustrated in Table 1(b) to the key-term dictionary.

In a similar process, the directory names d_l's are inspected and manually mapped to subject classes s_m's. In this case, an entire directory of images/videos corresponding to a particular topic and is mapped to one or more subject classes. For example, the directory d_l = "gallery/space/planets/saturn" is mapped to subject s_m = "astronomy/planets/saturn." Similar to the process for key-term extraction, the system computes a histogram $\{d_l : f_l\}$ of directory names and presents it for manual inspection. The directories that sufficiently group images and videos related to particular topics are then mapped to the appropriate subject classes.

In Section 6.1, we demonstrate that these methods of key-term and directory name extraction coupled with subject mapping provide excellent performance in classifying the images and videos by subject. We also hope that by incorporating some results of natural language processing [7], in addition to using visual features, we can further improve and automate the subject classification process.

3.4 Catalog Database

As described above, each retrieved image and video is processed and the following information tables are populated:

```
IMAGES    – IMID URL NAME FORMAT WIDTH HEIGHT FRAMES DATE
TYPES     – IMID TYPE
SUBJECTS  – IMID SUBJECT
TEXT      – IMID TERM
FV        – IMID COLOR-HISTOGRAM
```

where the special (non-alphanumeric) data types are given as follows:

`TYPE`	$\in$	{Color photo, Color graphic, Video, B/w image, Gray image}
`SUBJECT`	$\in$	{Subject classes from taxonomy $\{\widehat{s_m}\}$, partially depicted in Figure 3}
`COLOR-HISTOGRAM`	$\in$	$\mathcal{R}^{166}$ (166-bin histogram).

The automated assignment of `TYPE` to the images and videos using visual features is explained in Section 5.2. Queries on the database tables: `IMAGES`, `TYPES`, `SUBJECTS` and `TEXT` are performed using standard relational algebra. For example, the query: `Give me all records with TYPE` = "video", `SUBJECT` = "news" and `TERM` = "basketball" can be carried in SQL as follows:

SELECT `IMID`
FROM `TYPES, SUBJECTS, TEXT`
WHERE `TYPE =` "video" **AND** `SUBJECT =` "news" **AND** `TERM =` "basketball".

However, content-based queries, which involve table `FV`, require special processing, which is discussed in more detail in Sections 4.2 and 5.

4 Search, Browse and Retrieval

To search for images and videos, the user issues a query which extracts items from the catalog. The user may initiate the search by entering a term t_k or by selecting a subject s_m directly.

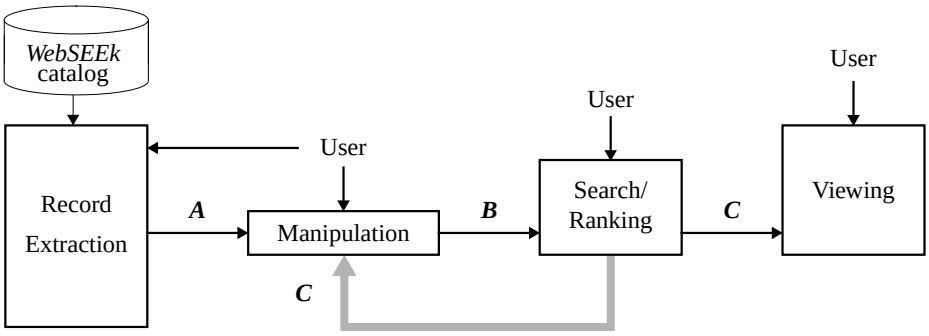
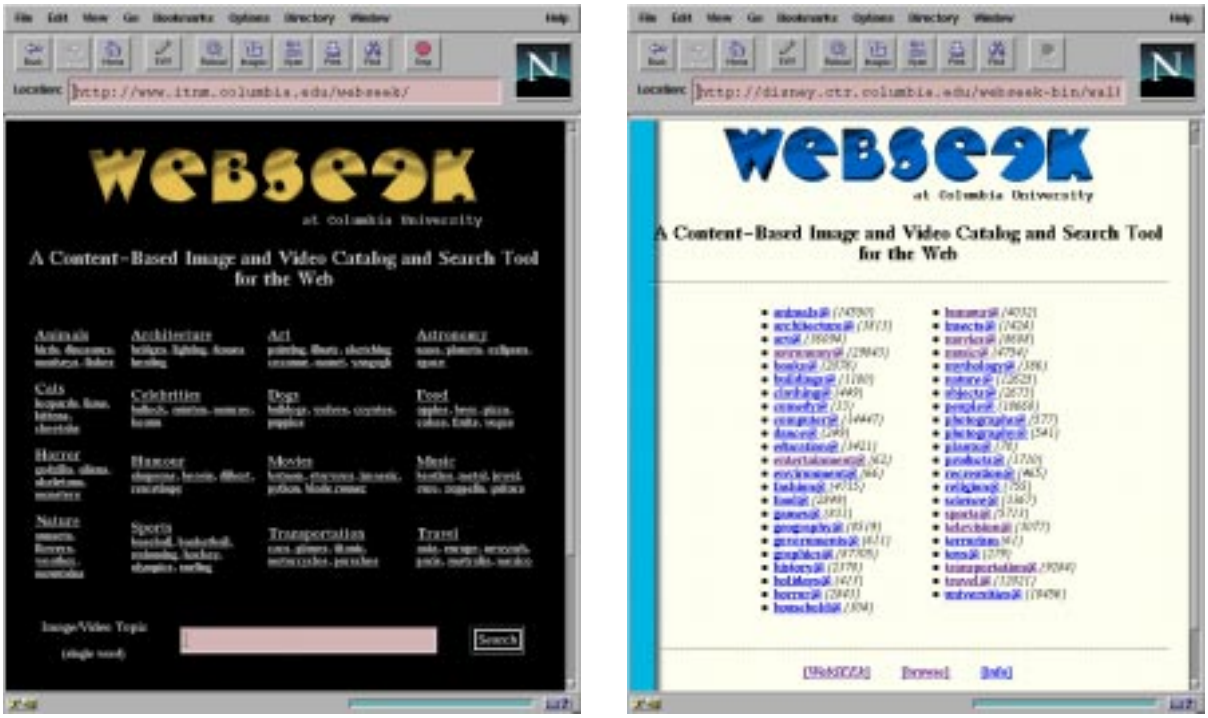


Figure 4: Search, retrieval and search results list manipulation processes.

The overall search process and model for user-interaction is depicted in Figure 4. As illustrated, a query for images and videos produces a search results list, list **A**, which is presented to the user. For example, Figure 6(a) illustrates the search results list for a query for images and videos related to "nature", that is, **A** = Query(`SUBJECT` = "nature"). The user may manipulate, search or view **A**. The user-interface and main search screens are illustrated in Figure 5(a) and (b).

4.1 Search Results List Manipulation

After possibly searching and/or viewing the search results list, it may be fed-back to the manipulation module as list **C**, as illustrated in Figure 4. The user manipulates **C** by adding or removing records. This is done by issuing a new query that creates a second, intermediate list **A**. The user then generates a new search results list **B** by selecting one of the following manipulations on **C** using **A**, for example, define **C** = Query(`SUBJECT` = "nature") and **A** = Query(`TERM` = "sunset"), then



(a) (b)

Figure 5: Main screens for (a) searching by selecting from several subjects or by text, (b) subject navigation.

union: $\mathbf{B} = \mathbf{A} \cup \mathbf{C}$, i.e., $\mathbf{B}$ = Query(TERM = "sunset" or SUBJECT = "nature"),
intersection: $\mathbf{B} = \mathbf{A} \cap \mathbf{C}$, i.e., $\mathbf{B}$ = Query(TERM = "sunset" and SUBJECT = "nature"),
subtraction: $\mathbf{B} = \mathbf{C} - \mathbf{A}$, i.e., $\mathbf{B}$ = Query(SUBJECT = "nature" and TERM $\neq$ "sunset"),
replacement: $\mathbf{B} = \mathbf{A}$, i.e., $\mathbf{B}$ = Query(TERM = "sunset"),
keep: $\mathbf{B} = \mathbf{C}$, i.e., $\mathbf{B}$ = Query(SUBJECT = "nature").

4.2 Content-based visual query

The user may browse and search the list $\mathbf{B}$ using both content-based and text-based tools. In the case of content-based searching, the output is a new list $\mathbf{C}$, where $\mathbf{C} \subseteq \mathbf{B}$ is an ordered subset of $\mathbf{B}$ such that $\mathbf{C}$ is ordered by highest similarity to the user's selected item. In the current system, list $\mathbf{C}$ is truncated to a default value of $\mathcal{N} = 60$ records, where $\mathcal{N}$ may be adjusted by the user. The search and browse operations may be conducted on the input list $\mathbf{B}$ or the entire catalog. In the first case, the user may browse the current search results list by selecting one of the items and instructing the system to reorder the list by highest similarity to the selected item.

For example, $\mathbf{C} = \mathbf{B} \simeq \mathbf{B}^{\text{sel}}$, where $\simeq$ means visual similarity, ranks list $\mathbf{B}$ in order of highest similarity to the selected item from $\mathbf{B}$. For example, the following content-based visual query:

$$\mathbf{C} = \text{Query}(\texttt{SUBJECT} = \text{"nature"}) \simeq \mathbf{B}^{\text{sel}}(\text{"mountain scene image"}),$$

ranks the "nature" images and videos in order of highest visual similarity to the selected "mountain scene image." Alternatively, the user can select one of the items in the current search results list $\mathbf{B}$ and then use it to search the entire catalog for similar items. For example, $\mathbf{C} = \mathbf{A} \simeq \mathbf{B}^{\text{sel}}$ ranks list $\mathbf{A}$, where, in this

case **A** is the full catalog, in order of highest visual similarity to the selected item from **B**. In the example illustrated in Figure 6(b), the query $\mathbf{C} = \mathbf{A} \simeq \mathbf{B}^{\mathrm{sel}}$("red race car"), retrieves the images and videos from the full catalog that are most similar to the selected image of a "red race car."

(a) (b)

Figure 6: (a) Search results for `SUBJECT` = "nature", (b) content-based visual query results for images/videos $\simeq$ "red race car".

4.3 Search Result List Views

The user has several options for viewing search results. Since the visual information requires more communication bandwidth than text, the user is given control in viewing and browsing the search results to enable them to be inspected quickly. The default view presents for each catalog record the small (approximately 96×96 pixels) icon for each image and video scene in addition to other relevant fields, see Figure 6(a) and (b). Alternatively, the user can select to eliminate the display of the icon altogether, in which case only the name of the image/video is displayed. Only $\mathcal{L} = 15$ records at a time are presented to the user in the view. The system gives the user controls to navigate the list by getting the *next*, *previous* and *top* $\mathcal{L}$ records. The user may conveniently select an item for full display, in which case, the system directs the user to the original *URL* of the image or video.

5 Content-based Techniques

The system provides tools for content-based searching for images and videos using color histograms generated from the visual scenes. We adopted color histograms in the system prototype in order to utilize a domain-independent approach. The content-based techniques developed here for indexing, searching and navigation can be applied, in principle, to other types of features and application domains.

The color histograms describe the distribution of colors in each image or video. We define the color histograms as discrete, 166-bin, distributions in a quantized HSV color space [8]. The system computes a

color histogram for each image and video scene, which is used to assess its similarity to other images and video scenes. The color histograms are also used to automatically assign the images and videos to type classes using Fisher discriminant analysis, as described in Section 5.2.

5.1 Color Histograms Similarity

The histogram dissimilarity function measures the weighted dissimilarity between histograms. For example, the quadratic distance between a query histogram $\mathbf{h}_q$ and a target histogram $\mathbf{h}_t$ is given by:

$$d_{q,t} = (\mathbf{h}_q - \mathbf{h}_t)^t \mathbf{A} (\mathbf{h}_q - \mathbf{h}_t), \tag{2}$$

where $\mathbf{A} = [a_{i,j}]$ is a symmetric matrix and $a_{i,j}$ denotes the similarity between colors with indexes i and j such that $a_{i,i} = 1$. Note that the histograms are normalized such that $||\mathbf{h}|| = 1$, where $||\mathbf{h}|| = \sqrt{\sum_{m=0}^{M-1} h[m]^2}$.

In order to achieve high efficiency in the color histogram query process, we decompose the color histogram quadratic formula. This provides for both efficient computation and indexing. By defining $\mu_q = \mathbf{h}_q^t \mathbf{A} \mathbf{h}_q$, $\mu_t = \mathbf{h}_t^t \mathbf{A} \mathbf{h}_t$ and $\mathbf{r}_t = \mathbf{A}\mathbf{h}_t$, the color histogram quadratic distance is given as

$$d_{q,t} = \mu_q + \mu_t - 2\mathbf{h}_q^t \mathbf{r}_t. \tag{3}$$

By partitioning vector $\mathbf{r_t}$ into elements $r_t[m]$'s, the distance function can be approximated to arbitrary precision by setting τ in

$$d_{q,t} - \mu_q = \mu_t - 2 \sum_{\forall m \text{ where } h_q[m] \geq \tau} h_q[m] r_t[m]. \tag{4}$$

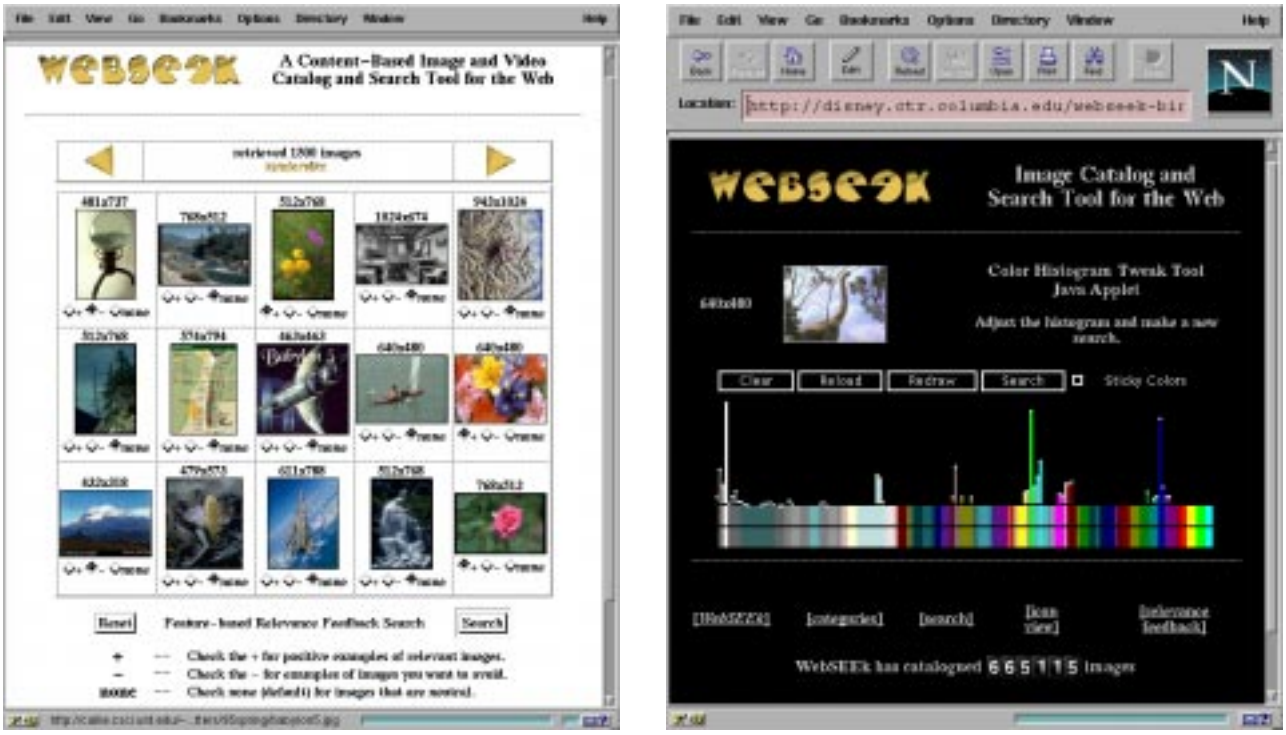
That is, any query for the most similar color histogram to $\mathbf{h}_q$ may be easily processed by accessing individually μ_t and $r_t[m]$'s, where $m \in 1 \ldots M$. Notice also that μ_q is a constant of the query. The closest color histogram $\mathbf{h}_t$ is given as the one that minimizes $\mu_t - 2 \sum_{\forall m \text{ where } h_q[m] \geq \tau} h_q[m] r_t[m]$. By using the efficient computation described in Eq. 4, we are able to greatly reduce the query processing time, as demonstrated in Section 6.3.

5.2 Automated Type Assessment

By training on samples of the color histograms of images and videos, we developed a process of automated type assessment using Fisher discriminant analysis. Fisher discriminant analysis constructs a series of uncorrelated linear weightings of the color histograms that provide for maximum separation between training classes. In particular, the linear weightings are derived from the eigenvectors of the matrix given by the ratio of the between-class to within-class sum-of-square matrices for K classes [9]. New color histograms, $\mathbf{h}_n$ are then automatically assigned to nearest type class k where

$$[\mathbf{T}(\mathbf{h}_n - \mathbf{m}_k)]^2 \leq [\mathbf{T}(\mathbf{h}_n - \mathbf{m}_i)]^2, \ \ \forall i \neq k, \tag{5}$$

and where $\mathbf{T}$ is the matrix of eigenvectors derived from the training classes and color histograms, and $\mathbf{m}_i$ is the mean histogram for class i. In Section 6.2, we show that this approach provides excellent automated classification of the images and videos into several broad type classes. We hope to further increase the number of type classes and improve the classification performance by incorporating other visual features into the process.



(a) (b)

Figure 7: (a) Relevance feedback search allows user to select both positive and negative examples, (b) histogram manipulation allows user to add and remove colors and adjust the color distribution for the next query.

5.3 Relevance Feedback

The user can best determine from the results of a query which images and videos are relevant and not relevant. The system can use this information to reformulate the query to better retrieve the images and videos the user desires [10]. Using the color histograms, relevance feedback is accomplished as follows: let $I_r = \{\text{relevant images/videos}\}$ and $I_n = \{\text{non-relevant images/videos}\}$ as determined by the user. The new query vector $\mathbf{h}_q^{k+1}$ at round $k+1$ is generated by

$$\mathbf{h}_q^{k+1} = ||\alpha \mathbf{h}_q^k + \beta \sum_{i \in I_r} \mathbf{h}_i - \gamma \sum_{j \in I_n} \mathbf{h}_j||, \tag{6}$$

where $||\cdot||$ indicates normalization. The new images and videos are retrieved using $\mathbf{h}_q^{k+1}$ and the distance metric in Eq. 4. One formulation of relevance feedback assigns the values $\alpha = 0$, and $\beta = \gamma = 1$, which weights the positive and negative examples equally. The process of selecting the example images for content-based relevance feedback searching is illustrated in Figure 7(a). A simpler form of relevance feedback allows the user to select only one positive example in order to iterate the query process. In this case, $\alpha = \gamma = 0$, $\beta = 1$, $|I_r| = 1$ and $|I_{nr}| = 0$ gives the new query vector directly from the selected image/video's color histogram, $\mathbf{h}_{I_r}$ as follows,

$$\mathbf{h}_q^{k+1} = \mathbf{h}_{I_r}. \tag{7}$$

5.4 Histogram Manipulation

The system also provides a tool for the user to directly manipulate the image and video color histograms to formulate the search. The modified histogram is then used to conduct the next search. The new query histogram $\mathbf{h}_q^{k+1}$ is generated from a selected histogram $\mathbf{h}_s$ by adding or removing colors, which are denoted in the modifications histogram $\mathbf{h}_m$, as follows,

$$\mathbf{h}_q^{k+1} = ||\mathbf{h}_s + \mathbf{h}_m||. \tag{8}$$

Using the histogram manipulation tool, illustrated in Figure 7(b), the user may select one of the images or videos from the results and display its histogram. The user can modify the histogram by adding or removing colors. The modified histogram is then used to conduct the next search.

6 Evaluation

In the initial trials, the system has catalogued 513,323 images and videos from 46,551 directories at 16,773 Web sites. The process required several months, and was performed simultaneously with the development of the user application. Various information about the catalog process is summarized in Table 2. Overall the system has catalogued over 129 Gigabytes of visual information. The local storage of information, which includes coarse versions of the data and color histogram feature vectors, requires approximately 2 Gigabytes.

6.1 Subject Classification Evaluation

As indicated in Table 2, the catalog process assigned 68.23% of the images and videos into subject classes using automated mapping for key-terms and semi-automated mapping for directory names. We assessed the subject classification rates for several classes, which is summarized in Table 3(a). The overall performance is excellent, $\sim 92\%$ classification precision. For this assessment, as illustrated in Table 3(a),

number of images/videos catalogued	513,323
number of Web sites providing the images and videos	16,773
number of distinct Web directories	46,551
% of catalog is black & white/gray-scale	14.15%
% of catalog is videos	1.05%
% of images and videos classified into subject classes	68.23%
size of subject taxonomy (# classes)	2128
size of key-term dictionary (# terms)	1703
number of directory name to subject mappings	1612

Table 2: Cataloging of 513,323 images and videos from the Web.

we chose the classes at random from the subject taxonomy of 2128 classes. We established the ground-truth by manually verifying the subject of each image and video in the test sets.

We observed that errors in classification result from several occurrences: (1) key-terms used out of context by the publishers of the images or videos, (2) reliance on some ambiguous key-terms, i.e., "madonna" and (3) reliance on key-terms extracted from directory names. For example, in Table 3(a), the precision of subject class "animals/possums" is low because five out of the nine items are not images or videos of possums. These items were classified incorrectly because the key-term "possum" appeared in the directory name. While some of the images in that directory depict possums, others depict only the forests to which the possum are indigenous. When viewed outside of the context of the "possum" web site, the images of forests should not be assigned to the class "animals/possums."

6.2 Type Classification Evaluation

We assessed the precision of the automated type classification system, which is summarized in Table 3(b). For this evaluation, both the Training and Test samples consisted of 200 images from each type class. We found the automated type assessment for these five simple classes is quite satisfactory, overall $\sim 95\%$ rate of successful classification. In future work, we will try to extend this system to include a larger number of classes, including new type classes, such as *Fractal* images, *Cartoons*, *Faces*, *Art paintings* and subject classes.

Subject	# sites	Count	Rate
art/illustrations	29	1410	0.978
entertainment/humour/cartoons/daffyduck	14	23	1.000
animals/possums	2	9	0.444
science/chemistry/proteins	7	40	1.000
nature/weather/snow/frosty	9	13	1.000
food	403	2460	0.833
art/paintings/pissarro	3	54	1.000
entertainment/music/mtv	15	87	0.989
horror	366	2454	0.968

Type	Rate
Color photo	0.914
Color graphic	0.923
Gray image	0.967
B/w image	1.000

Table 3: Rates of correct (a) Subject classification (precision) for random set of classes and (b) automated type classification.

6.3 Efficiency

Another important factor in the image and video search system is the speed at which user operations and queries are performed. In particular, as the archive grows it is imperative that queries do not take so long that they inhibit the user from effectively using the system. In the initial system, the overall efficiency of various database manipulation operations is excellent, even on the large catalog. In particular, the good performance of the content-based visual query tools is given by the strategy for indexing the 166-bin color histograms described in Section 5.1. For example, the system identifies the $\mathcal{N} = 60$ most similar visual scenes in the catalog of 513,323 images and videos to a selected query scene in only 1.83 seconds.

7 Summary and Future Work

We introduced a new robust system that provides the essential function of cataloging the visual information on the Web. The system automatically collects the images and videos and catalogs them using both the textual and visual information. This web application is very easy to use and provides great flexibility and functionality for searching and browsing for images and videos. In the initial implementation, the system has catalogued more than one half million images and videos.

In future work, we will incorporate other visual dimensions, such as texture, shape and spatial layout [11], to enhance the content-based components of the system. In particular, we are applying our recent *VisualSEEk* [12] system for joint spatial and feature querying to this application. We are also incorporating automated techniques for detecting faces [13] and text in images and videos.

We also plan to investigate new techniques for exploiting text and visual features independently and jointly to improve the process of cataloging the images and videos and automatically mapping them into subject and type classes. For example, better utilization of the text information in the parent Web pages may provide more information about the images and videos [7]. In addition, several recent approaches for learning from visual features appear promising for detecting homogeneities within subject classes and for improving the automated classification system. Finally, we will further expand and define the image and video subject taxonomy.

8 Acknowledgments

This work was supported in part by the National Science Foundation under a CAREER award (IRI-9501266), IBM under a 1995 Research Partnership (Faculty Development) Award, and sponsors of the ADVENT project of Columbia University. The authors would like to thank Kazi Zaman, Dragomir Radev, Al Aho and Kathleen McKeown for their valuable input on this project.

References

[1] G. S. Jung and V. N. Gudivada. Autonomous tools for information discovery in the world-wide web. Technical Report CS-95-01, School of Electrical Engineering and Computer Science, Ohio University, Athens, OH, 1995.

[2] S. Sclaroff. World wide web image search engines. In *NSF Workshop on Visual Information Management*, Cambridge, MA, June 1995.

[3] J. R. Smith and S.-F. Chang. Searching for images and videos on the World-Wide Web. Technical Report CU/CTR 459-96-25, Columbia University, August 1996.

[4] M. Koster. Robots in the web: threat or treat? *ConneXions*, 9(4), April 1995.

[5] J. Meng, Y. Juan, and Shih-Fu Chang. Scene change detection in a MPEG compressed video sequence. In *Proceedings of the IS&T/SPIE Symposium on Electronic Imaging: Science & Technology*, San Jose, CA, February 1995.

[6] I. H. Witten, A. Moffat, and T. C. Bell. *Managing Gigabytes: compressing and indexing documents and images*. Van Nostrand Reinhold, New York, NY, 1994.

[7] E. J. Guglielmo and N. C. Rowe. Natural-language retrieval of images based on descriptive captions. In *ACM Trans. Info. Systems*, volume 14, pages 237 – 267, July 1996.

[8] J. R. Smith and S.-F. Chang. Tools and techniques for color image retrieval. In *Symposium on Electronic Imaging: Science and Technology – Storage & Retrieval for Image and Video Databases IV*, volume 2670, San Jose, CA, February 1996. IS&T/SPIE.

[9] K. Fukunaga. *Introduction to Statistical Pattern Recogntion*. Harcourt Brace Javanovich, 1990.

[10] J. J. Rocchio Jr. Relevance feedback in information retrieval. In Gerard Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, chapter 14, pages 313 – 323. Prentice-Hall, Englewood Cliffs, New Jersey, 1971.

[11] J. R. Smith and S.-F. Chang. Local color and texture extraction and spatial query. In *Proc. Int. Conf. Image Processing*, Lausanne, Switzerland, September 1996. IEEE.

[12] J. R. Smith and S.-F. Chang. VisualSEEk: a fully automated content-based image query system. In *Proc. ACM Intern. Conf. Multimedia*, Boston, MA, November 1996. ACM.

[13] H. Wang and S.-F. Chang. Automatic face region detection in MPEG video sequences. In *Electronic Imaging and Multimedia Systems, SPIE Photonics China '96*, November 1996.