Querying by Color Regions using the VisualSEEk Content-Based Visual Query System

John R. Smith and Shih-Fu Chang

Center for Image Technology for New Media
and Department of Electrical Engineering
Columbia University
530 West 120th St.
New York, NY 10027
USA
1 (212) 854-2380
{*jrsmith,sfchang*}@*ee.columbia.edu*

Abstract

VisualSEEk is a highly functional image database manipulation system that provides advanced user tools for searching, browsing and retrieving images. VisualSEEk is distinct from other content-based image query systems in that the user may query for images using both the visual properties of regions and their spatial layout. Furthermore, the image analysis for region extraction is fully automated. VisualSEEk uses a novel system for region extraction and representation based upon color sets. Through a process of color set back-projection, the system automatically extracts salient color regions from images. This paper describes the implementation of the color query system and examines its role for content-based visual query in image and video databases. The next phases of the VisualSEEk implementation will provide additional tools for querying by texture, shape, embedded text and motion features.

1. Introduction

This paper describes the design, implementation and operation of the VisualSEEk content-based visual query system. VisualSEEk is a new content-based image query system that enables querying by both the visual properties of image regions and their spatial layout. VisualSEEk is easy to use and provides power and flexibility in the expression of visual queries. As the primary means for querying, the system utilizes visual features that are extracted from the images and videos using automated scene analysis. The integration of feature-based similarity and spatial querying relies on the representation of color regions

by color sets. Color sets provide for a convenient system of region extraction through back-projection [Smith and Chang, *SPIE*, 1996]. In addition, the color set distance function can be decomposed into terms that are easily indexed. This allows for the efficient computation of color set distance and the indexing of color regions. As a result, the system decomposes unconstrained images into near-symbolic images that lend to efficient spatial query, as depicted in Figure 1.

| (a.) | (b.) | (c.) |
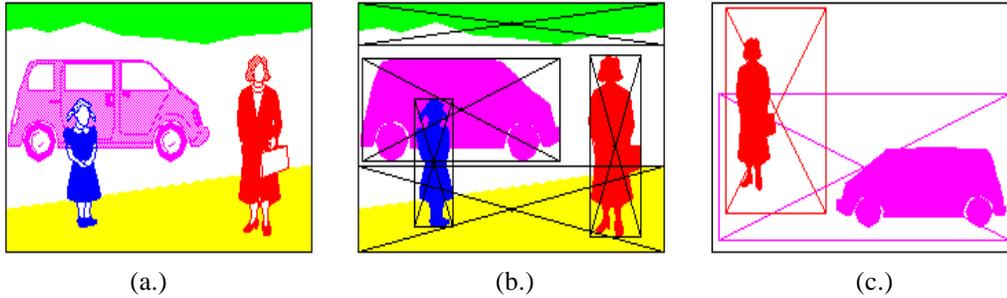
Figure 1 Image decomposition into localized color regions (a.) original target image, (b.) decomposed target image and (c.) sample query image.

In this paper we briefly describe the processes for extraction, representation and spatial query of the color regions. We also describe other components of VisualSEEk such as the Give Me More color search tools and text-based search and annotation tools. These components also have critical roles in the query system. For example, clustering the images and videos using visual features greatly accelerates the annotation process. Finally, we demonstrate the query tools and show some example queries. Color is just one visual modality we are investigating. We will soon integrate into the query system other visual features such as texture, shape, motion and embedded text.

1.1.   Image and Video Storage and Retrieval Systems

An image and video storage and retrieval (IVSR) system manages, stores and provides tools for search and retrieval of images and videos. Applications of IVSR include on-line stock photography, Video-On Demand [Chang, et. al., 1994], video browsing [Zhong, et. al., 1996] and video editing [Meng and Chang, 1996]. There are some great technical challenges in improving access to archived visual information. First, current search tools are primitive. Typically, the IVSR system utilizes textual descriptions or keywords for querying for visual

items. However, producing meaningful annotations for an entire archive of visual data requires great human effort. Any set of text descriptions cannot characterize the content of the images and videos sufficiently or consistently. Second, the enormity of the visual data prohibits the user from visually inspecting all but small portions of the archive at a time. Therefore, the ability to find the desired images and videos depends primarily on the capabilities of the query tools the system provides. Finally, traditional ISVR systems do not enable the user to retrieve items using queries of the type: "give me all images that look like this one." Rather, content-based visual query tools provide this type of enhanced operation.

1.2.   Content-based Visual Query Systems

Content-based visual query (CBVQ) tools enhance the operation of IVSR systems by providing the means for conducting visual searches. The user may search for images in several ways, (1) by image example, (2) by sketch or (3) by features, such as color and texture. Thus far, research on CBVQ has focused on color, texture and shape features because these are most natural for the user and can be represented effectively by computer. For example, the IBM Query by Image Content (QBIC) project [Faloutsos, et. al., 1994] provides tools for querying by color, texture and shape. Other research systems such as MIT's Photobook system [Pentland, et. al., 1994] and the Virage system [Bach, et. al., 1996] also provide CBVQ tools for searching for images by color, texture and shape. However, these systems have neglected an important aspect of visual perception -- spatial information and spatial relationships.

There are still many unresolved research issues in CBVQ, such as, (1) what visual features are best, (2) how should they be extracted, coded and compared, (3) what are the proper searching techniques and indexing structures and (4) how should the user best browse and search for images. Furthermore, particular applications require custom solutions. For example, searching through a medical image archive using medical artifacts requires a different approach and different visual features than a satellite image system using terrain features.

1.3. The VisualSEEk CBVQ System

The goal of the VisualSEEk project is to implement a CBVQ system that provides the means for efficient and effective retrieval of images and video [Chang, et. al., 1996]. The VisualSEEk project has emphasized several unique objectives in order to improve functionality: (1) automated extraction of localized regions and features [Smith and Chang, *SPIE*, 1996] (2) querying by both feature and spatial information [Smith and Chang, *SPIE*, 1996], (3) extraction from compressed data [Chang and Smith, 1995], (4) fast indexing and retrieval and (5) highly functional user tools. We developed the VisualSEEk client application in the Java language to allow for maximum functionality, client platform independence and accessibility on the World Wide Web. Currently, the VisualSEEk system provides a test-bed of 12,000 color images.

2. System Requirements and Design

The overall system consists of several components: the VisualSEEk user tools, the query server, the image and video retrieval server, the image and video archive, the meta-data database and the index files, as illustrated in Figure 2.
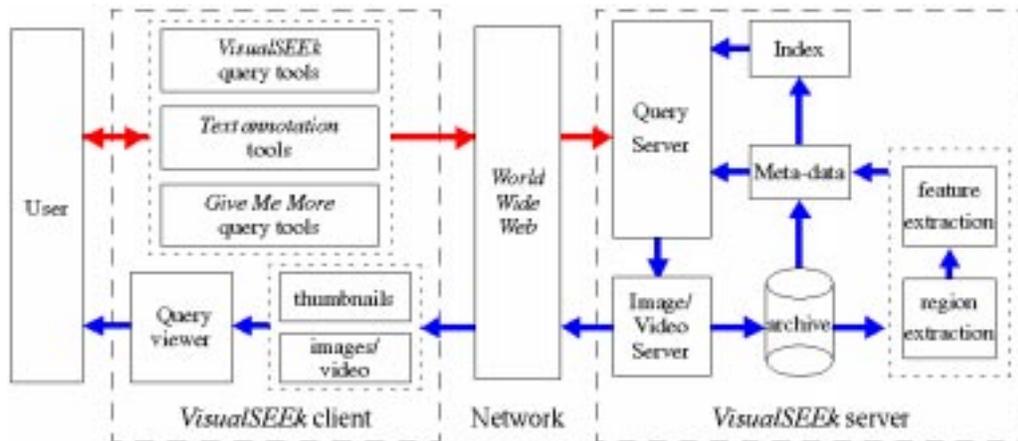
Figure 2 The VisualSEEk system.

2.1. System Architecture

The VisualSEEk system consists of three applications: (1) the client application, (2) the network and communication application, and (3) the server application, as illustrated in Figure 3. The client application consists of a suite of Java applets that execute within the World Wide Web browser. The VisualSEEk applet collects the query from the user and sends the query string to the server. The Hypertext Transfer Protocol (HTTP) client-server system handles all communication across the World Wide Web. The Common Gateway Interface (CGI) component of the HTTP executes the server program on the VisualSEEk query server machine.
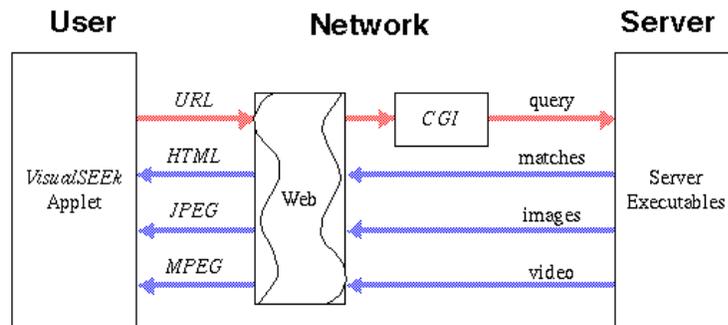
Figure 3 VisualSEEk system overview.

2.2. User Application

The user application collects the query from the user and generates the query string. The query tools allow the selection of region properties and the assignment of the spatial layout of regions. For example, one tool coordinates the placement of the color regions on the query grid. Another allows for selection of color. The client application packages the user's query into the query string which it sends to the server.

2.3. Server Application

The server application receives the query string, executes the query and returns the results to the user. The server first parses the query string and processes the query's color selection. It converts the color parameters into a corresponding

color in Hue-Saturation-Value (*HSV*) color space. Next, it returns to the user a mock-up image that depicts the received query. Then, a query program finds the best matches to the user's query. This program produces Hypertext Markup Language (HTML) code that displays the results of the query to the user. The HTML output includes thumbnail images to depict the images and videos found in the query. The output also indicates the distance scores for each matched item. When the user selects a thumbnail image the system retrieves the corresponding image or video from the image and video server. The system also allows the user to search again by image example using the returned images.
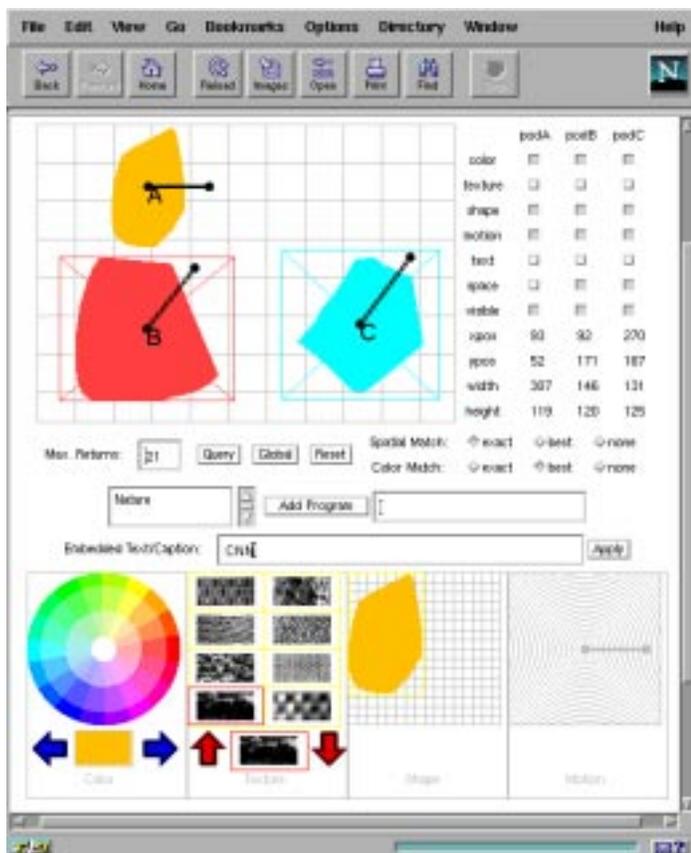
Figure 4 VisualSEEk Java user interface

2.4. Color Region Queries

The user graphically constructs joint color/spatial queries using the VisualSEEk query tools illustrated in Figure 4. The user sketches the regions, positions them on the query grid and assigns them properties of color, size and absolute location. The user may also assign boundaries for location and size of the regions. The spatial relationships between regions are diagrammed in VisualSEEk by connecting regions. In this way, the interface provides for construction of queries that include region features and combinations of both absolute and relative placement of the regions. For example, Figure 5(a) depicts a query that specifies three regions – blue, green and light blue with no constraint on their spatial arrangement. Figure 5(b) illustrates the results of the query which has found images of underwater scenes and nature with blue skies and trees.

(a)   (b)
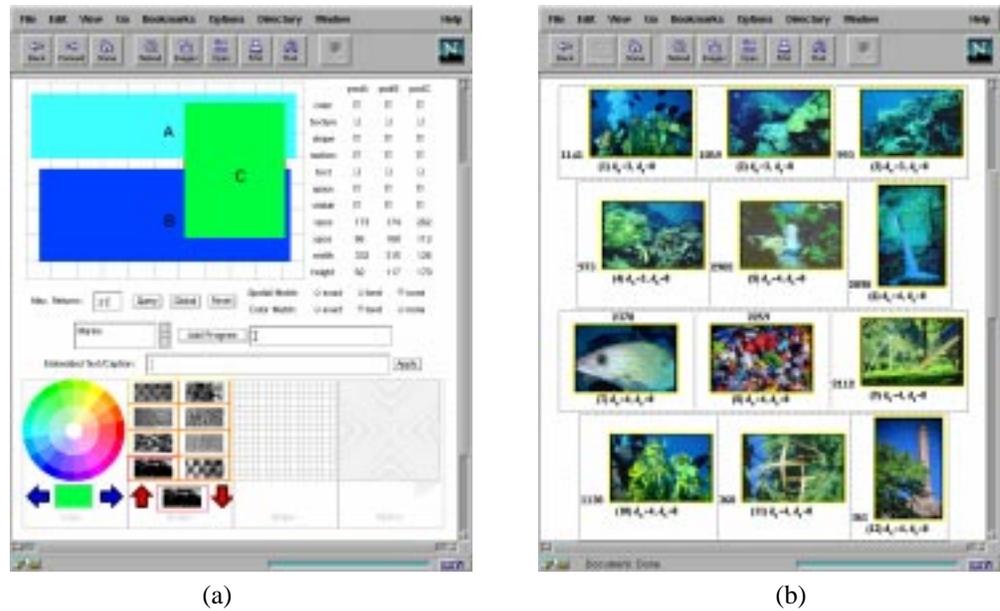
Figure 5 A sample query that specifies three colors, (a) query configuration in VisualSEEk and (b) query results.

2.5.   Give Me More Queries

After the system finds the matches and returns the thumbnail images, as in Figure 5(b), the user has several options: (1) select items for full retrieval and viewing, (2) modify the query and try again and (3) select items for Give Me More queries. In a Give Me More query, VisualSEEk searches the database for the best matches to the selected item. The Give Me More query tools allow the user to select from several global similarity metrics such as: global color, color regions, global texture, and joint color and texture. Figure 6 illustrates an example Give Me More query using global color features of a selected image from Figure 6(a). The images displayed in Figure 6(b) have the most similar color content to selected image.

| (a) | (b) |
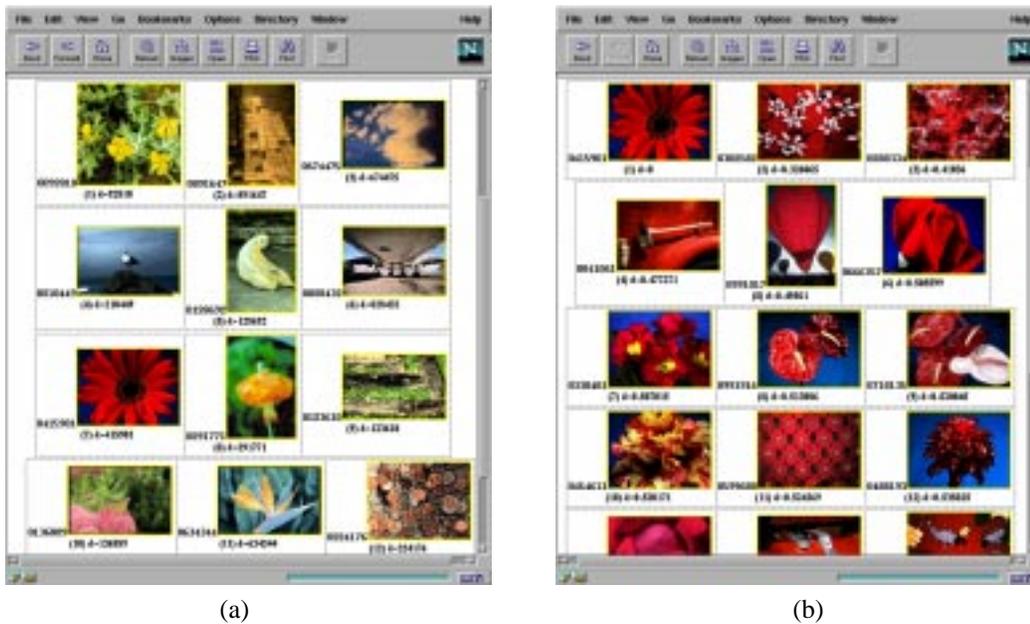
Figure 6 Give Me More query (a) images selected at random from the database, (2) results of a Give Me More query using global color of the image in (a) third down on the left.

2.6.   Image and Video Annotation

The VisualSEEk system also provides tools for annotating scenes individually and in groups. The annotation tool, Figure 7(a), allows the user to easily assign

text to images and videos. The user adds new annotations by entering keywords into the entry box corresponding to the image or video. VisualSEEk also provides a system for the group annotation of images. By clustering items from the database by their visual features, the user may annotate them in groups, which significantly reduces the time and effort required to annotate the archive. For example, the first image, which depicts an ocean scene, see Figure 7(a), is selected for "power annotation." The system searches the database for similar images and returns them to the user, Figure 7(b). Many of the returned images have a similar subject to the selected image and also depict ocean scenes. The user may now assign the annotations, as appropriate, to the group of images.
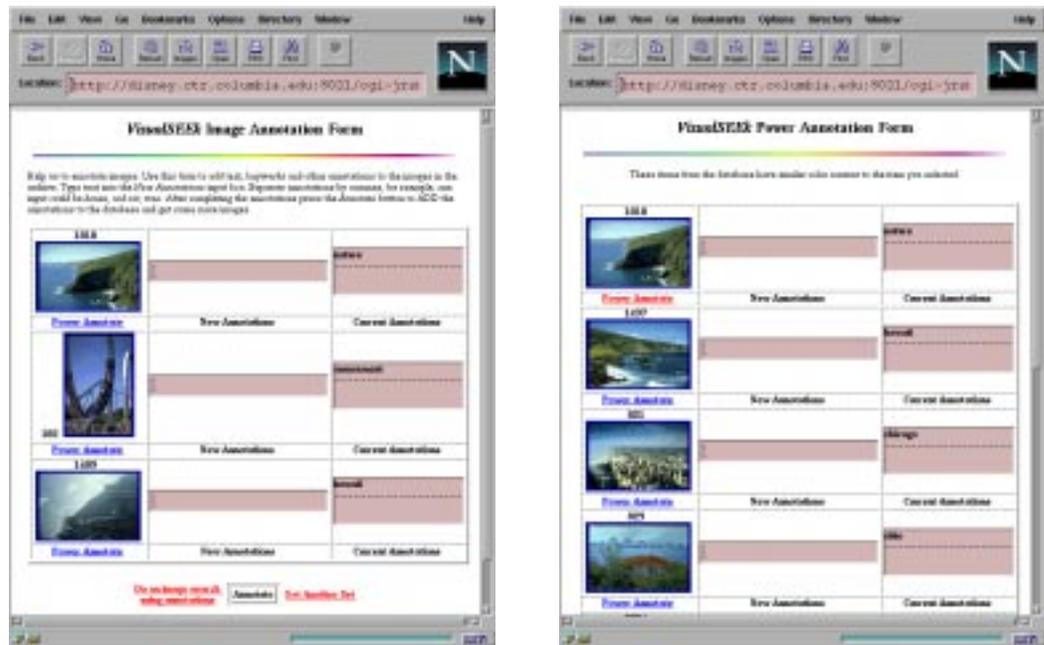
(a)   (b)

Figure 7 Annotation (a) individual image annotation (b) "power" annotation allows for group annotation of similar images.

2.7.   Integrated Text/Image Feature Searching

Text-based searching is still an important component of CBVQ systems. Although text alone is not always sufficient, text-searching combined with

CBVQ improves query capabilities. For example, given that some of the images and videos are annotated, a user may find some matches in a text-based search. If the text-based search is partially successful, the user can select some of the resulting matches to seed visual feature-based queries. For example, say that only two of the many images of lions from the database have been annotated with the keyword "lion." By conducting a text-based search using "lion," the system finds both of the matches, see Figure 8(a). However, many images of lions remain in the archive and are not retrieved. By selecting one of the returned lion images and by issuing a Give Me More query based upon the visual features of the selected image, the system finds many of the remaining images of lions, see Figure 8(b).

(a)   (b)
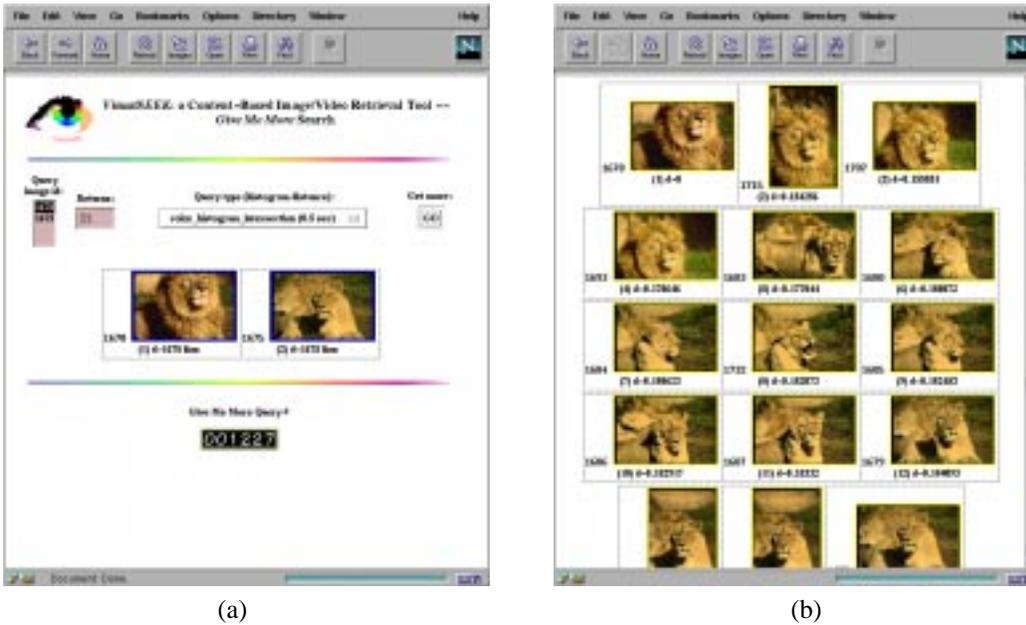
Figure 8 Integrated text/image-feature searching (a) text-based search using keyword "lion" yields only two matches, (b) feature-based search using a lion image from (a) finds many images of lions.

3.   Querying Using Spatially Localized Color Regions

As discussed already, there are two approaches for querying by color: by regional color and by global color. Regional color corresponds to spatially

localized colored regions within the scenes. Global color corresponds to the overall distribution of color within the entire scene. This next section describes the process for the automated extraction and query of spatially localized color regions, which uses color sets. Section 4.3 describes in more detail the process for querying using global color.

Color sets provide a compact and easily indexed representation of regional color content [Smith and Chang, *SPIE,* 1996]. The extraction and query of color sets is analogous the process of file inversion for search and retrieval of textual document [Witten, et. al, 1994]. The goal is to efficiently index the color regions. A binary color set back-projection algorithm extracts the color regions automatically. In short, VisualSEEk extracts color regions in a process that (1) transforms, (2) quantizes and (3) filters the images and videos to emphasize prominent color regions. The back-projection algorithm extracts the prominent color regions, which the system represents and indexes by color set value.

3.1. Color Representation and Back-Projection

Color sets provide an efficient alternative to color histograms for representation of color information. The following paragraphs define color sets and explain their relationship to color histograms.

3.1.1. Color Set Representation of Color Regions

Three dimensions of color can be defined and measured. For example, each image point can be represented as a 3-D vector $v_c = (r, g, b)$ in the *RGB* color space. The transformation T_c and quantization Q_c^M of the *RGB* color space reorganize and group the vectors v_c. The perceptually distinct colors are obtained by the sets of vectors that are mapped to different indices *M* as diagrammed in Figure 9. We define color sets as follows: let B_c^M be the *M* dimensional binary space such that each axis in B_c^M corresponds to one unique index value *m*. A color set is a binary vector in B_c^M which corresponds to a selection of colors $\{m\}$.
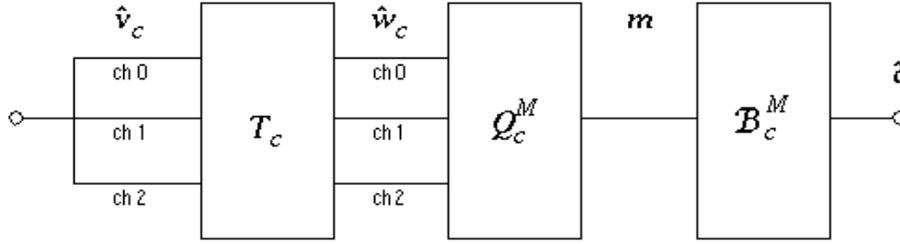
Figure 9 Generation of the binary color space from three color channels.

3.1.2. Color Set Example

For example, let T_c transform *RGB* to *HSV* and let Q_c^M where $M = 8$ quantize the *HSV* color space to 2 hues, 2 saturations and 2 values. The quantizer Q_c^M assigns a unique index *M* to each quantized *HSV* color. Then, B_c^8 is the eight dimensional binary space whereby each axis in B_c^8 corresponds to one of the quantized *HSV* colors. A color set $\daleth$ contains a selection from the eight colors. For example, the color set $\daleth = [10010100]$ corresponds to the selection of three colors, $m = 7$, $m = 4$ and $m = 2$ from the quantized *HSV* color space. In the VisualSEEk application, we chose T_c to transform *RGB* to *HSV* because *HSV* provides a breakdown of color into its most natural components: hue, saturation and intensity[1]. We chose the quantization Q_c^M to provide $M = 166$ represented colors as illustrated in Figure 10 [Smith and Chang, *SPIE,* 1996].
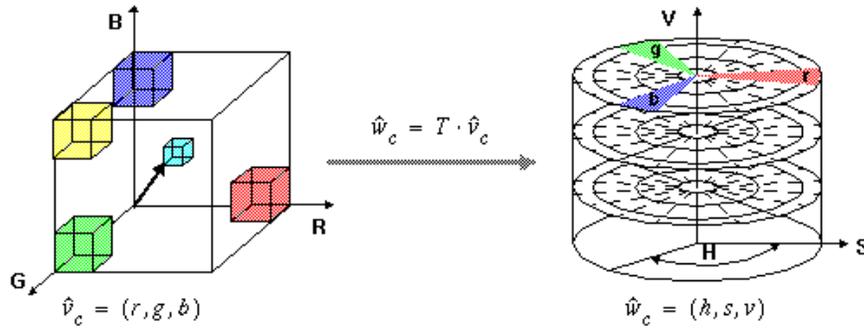
Figure 10 Transformation T_c from *RGB* to *HSV* and quantization gives 18 hues, 3 saturations, 3 values and 4 grays = 166 colors.

3.1.3. Color Set Back-Projection

The process of region extraction requires several stages: color set iteration, back-projection, thresholding and labeling. The color set iteration procedure selects candidate color sets to back-project onto the image by considering the content of the image histogram and the previously extracted color regions. The back-projection of a color set is accomplished as follows: given image $I[x, y]$ and binary color set $\vec{c}$, let $k = I[x, y]$, then generate bi-level image $B[x, y] = c[k]$. That is, $B[x, y]$ depicts the back-projection of color set $\vec{c}$. The bi-level image $B[x, y]$ is then filtered morphologically and analyzed to reveal spatially localized color regions [Smith and Chang, *SPIE*, 1996].

3.1.4. Color Similarity

The similarity between any two colors $m_q = (h_q, s_q, v_q)$ and $m_t = (h_t, s_t, v_t)$ is given by

$$a_{q,t} = 1 - \frac{[(v_q - v_t)^2 + (s_q \cos h_q - s_t \cos h_t)^2 + (s_q \sin h_q - s_t \sin h_t)^2]^{\frac{1}{2}}}{\sqrt{5}} \qquad \text{Eq. 1}$$

which is derived from the distance in the cylindrical *HSV* color space depicted in Figure 10. Single color similarity is used within the computation of the distance between color distributions.

3.1.5. Color Histograms

A color histogram corresponds to a distribution of colors. By transforming the three color channels of image $I[x, y]$ using transformation T_c and quantization Q_c^M where $(\vec{c})_{xy} = (I_R[x, y], I_G[x, y], I_B[x, y])$ the single variable color histogram is given by, where X and Y are the width and height of the image, respectively,

$$h[m] = \frac{1}{XY} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \begin{cases} 1 & \text{if } Q_c^M T_c(\vec{c})_{xy} = m \\ 0 & \text{otherwise.} \end{cases} \qquad \text{Eq. 2}$$

3.1.6.   Histogram Quadratic Distance

The QBIC project proposed the histogram quadratic distance for image retrieval [Niblack, et. al, 1993]. It measures the weighted similarity between histograms, which provides more desirable matching than "like-bin" only comparisons. The quadratic distance between histograms $\vec{h}_q$ and $\vec{h}_t$ is given by

$$d_{q,t}^{hist} = (\vec{h}_q - \vec{h}_t)^t \, A \, (\vec{h}_q - \vec{h}_t) \, ,$$
<div align="right">Eq. 3</div>

where $A = [a_{ij}]$ and a_{ij} denotes the similarity of colors with indexes i and j. By defining color similarity in *HSV* color space, a_{ij} is given by Eq. 1. Since the histogram quadratic distance computes the cross similarity between colors, it is computationally expensive.   Therefore, large applications require other strategies, such as pre-filtering to avoid exhaustive search [Hafner, et. al, 1995].

3.1.7.   Color Sets *vs.* Color Histograms

Color sets, as defined in Section 3.1.1, give only the selection of colors, whereas, a color histogram denotes the relative amounts of colors. A color set approximates a color histogram by thresholding the color histogram.   For example, given threshold τ_m for color *m*, a color set is obtained from (see also Figure 11),

$$c[m] = \begin{cases} 1 & if \ h[m] \geq \tau_m \\ 0 & otherwise. \end{cases}$$
<div align="right">Eq. 4</div>

The color set indicates only those colors that are found above threshold levels. It works well to represent regional color since (1) T_c and Q_c^M have been derived to give a complete set of distinct colors and (2) the region extraction is based upon the dominant colors in the regions [Smith and Chang, *SPIE*, 1996]. A color that is not well represented in a region -- for example, if it is below threshold τ_m -- is ignored. One good choice of threshold τ_m is $\tau_m = 1/\sigma_m^2$ where σ_m^2 is the variance of color *m* in a random sample of images.
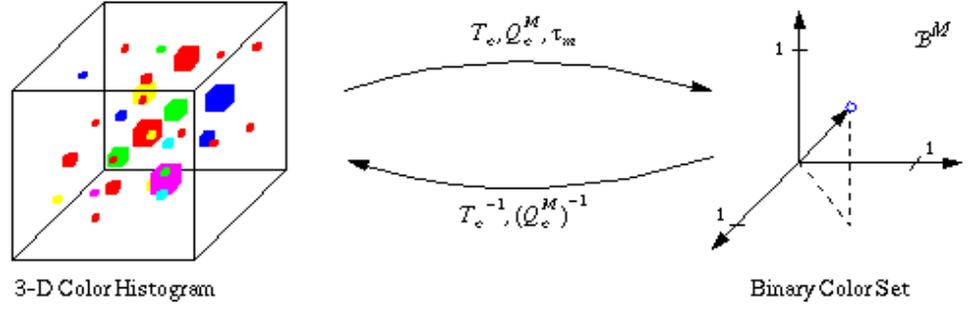
Figure 11 Relationship between 3-D color histogram and binary color sets.

3.1.8.   Color Set Distance

The histogram quadratic distance equation, Eq. 3, is also used for the distance between color sets. The distance between two color sets $\vec{c}_q$ and $\vec{c}_t$ is given by

$$d_{q,t}^{set} = (\vec{c}_q - \vec{c}_t)^t \, A \, (\vec{c}_q - \vec{c}_t). \qquad \text{Eq. 5}$$

However, given the binary nature of the color set representation, the computational complexity of the quadratic distance function can be reduced. We decompose the color set quadratic formula to provide for efficient computation and indexing. By defining $\mu_q = \vec{c}_q^t A \vec{c}_q$, $\mu_t = \vec{c}_t^t A \vec{c}_t$ and $\vec{R}_t = A \vec{c}_t$, and since A is symmetric, the color set quadratic distance is equivalent to

$$d_{q,t}^{set} = \mu_q + \mu_t - 2c_q^t \vec{R}_t. \qquad \text{Eq. 6}$$

Since $\vec{c}_q \in B^M$ ($\vec{c}_q$ is a binary vector),

$$d_{q,t}^{set} - \mu_q = \mu_t - 2 \sum_{\forall m \text{ where } c_q[m]=1} R_t[m]. \qquad \text{Eq. 7}$$

That is, any query for the most similar color set to $\vec{c}_q$ may be easily processed by storing and indexing individually μ_t and $R_t[m]$, where $m \in 1K\ M$. Notice also that μ_q is a constant of the query. The closest color set $\vec{c}_t$ is given as the one that minimizes $\mu_t - 2\sum_{\forall m \text{ where } c_q[m]=1} R_{t[m]}$.

4.   Query Examples and Evaluation

4.1.   Joint content-based/spatial queries

We now present some example joint content-based/spatial queries in VisualSEEk. In the first example, see Figure 12(a), the query (top) specifies the absolute location of a single region. The retrieved image (bottom) has the best match in the features (color and size) to the query region. In the next example, see Figure 12(b), the query specifies multiple regions. The retrieved image provides the best match in terms of the features and absolute spatial locations of the query regions. In the next example, see Figure 12(c), the query specifies the spatial relationships of regions. The retrieved image has three regions that best match the features of the query regions and their spatial relationship satisfies that specified in the query. Finally, see Figure 12(d), the query specifies both absolute and relative locations of regions. In this query, the match to the region positioned by absolute location (top left region in query image) considers both the features and location of this region. The match to the other regions (bottom two regions in query image) at first considers only the features of these regions. In the last stage of the query, the spatial relationship of the regions is evaluated to determine the match.

(a.)   (b.)   (c.)   (d.)
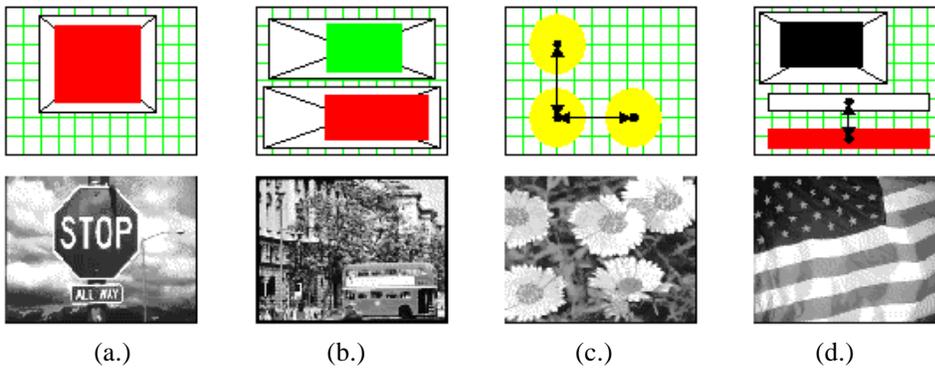
Figure 12   Example VisualSEEk queries, (a.) single region with absolute locations, (b.) two regions with absolute locations, (c.) multiple regions with relative locations, (d.) multiple regions with absolute and relative locations.

4.2. Color Region Query Evaluation

We evaluate the retrieval effectiveness of querying by color regions using recall and precision [Jones, 1981]. Prior to the trials, we assigned each image a relevance to the queries using a three-value relevance. As illustrated in Figure 13(a), when looking for particular color regions within images, the spatially localized color region query method performs better than global feature querying. In the global feature queries, the user picked a seed image from the collection that contained the desired regions. The results indicate that the VisualSEEk color region query tools provide an effective means for retrieving images based upon the color and spatial arrangement of regions.

4.3. Global Feature Querying

The global color features of images are compared using color histograms as defined in Eq. 2. Likewise, the similarity of images is computed using the color histogram similarity equation, Eq. 3. In general, image feature histograms provide a powerful representation for querying using global features. We have found histograms to provide for excellent representation of several modalities of visual features, including color, texture and joint color and texture. For example, to represent texture, VisualSEEk utilizes a texture histogram that is derived from spatial-frequency channels [Smith and Chang, *ICASSP*, 1996].

Querying by global features in particular situations is more desirable than querying by localized regions. For example, when spatial information is not important and the image background does not confound the similarity measurements, querying by global features is more appropriate than querying by regions. To illustrate, we evaluate the retrieval effectiveness of querying by both methods for images of sunsets. As illustrated in Figure 13(b), querying by local color has a lower precision initially because the system cannot distinguish between images of sunsets and images of orange and yellow flowers. On the other hand, querying using the global features of example sunset image gives higher precision than querying by color regions.

<table>
<tr><td>(a)</td><td>(b)</td></tr>
</table>
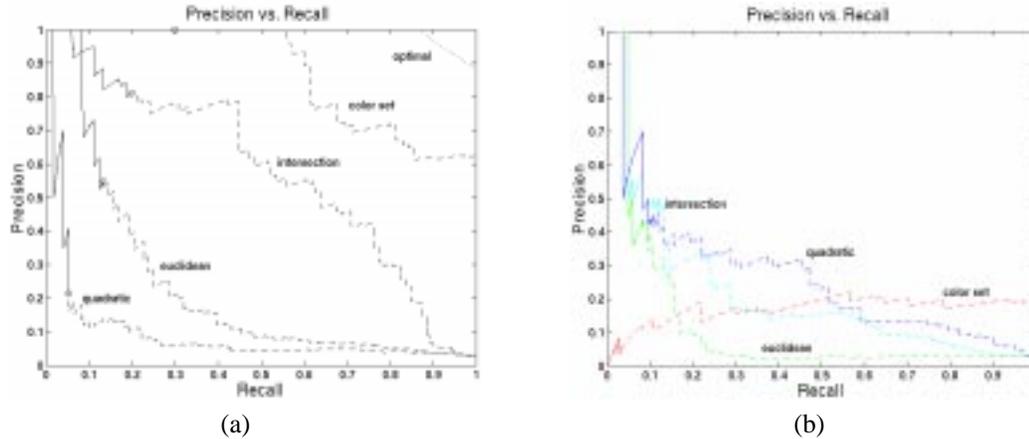
Figure 13 Retrieval effectiveness. (a) Query 1. Single color regions. (b) Query 2. Sunsets.

5.   Summary and Future Work

This paper presented the VisualSEEk content-based visual query system. VisualSEEk provides tools for querying by color and the spatial layout of color regions. The underlying representation for color regions is based upon binary color sets. VisualSEEk also provides tools for searching by image example. VisualSEEk also provides tools for annotating images and for searching the archive using text. In future work we will extend many of the search capabilities of VisualSEEk to include other visual modalities such as texture, shape, embedded text and motion.

6.   References

J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. C. Jain and C. Shu. Virage image search engine: an open framework for image management. In *Symposium on Electronic Imaging: Science and Technology – Storage & Retrieval for Image and Video Databases IV*, San Jose, CA, February 1996. IS&T/SPIE.

S.-F. Chang, D. Anastassiou, A. Eleftheriadis, J. Meng, S. Paek, S. Pejhan and J. R. Smith. Development of advanced image/video servers in the video on demand testbed.   In Proceedings, *Visual Signal Processing & Communications Workshop*, New Brunswick, NJ, Sept. 1994. IEEE.

S. F. Chang and J. R. Smith. Extracting multi-dimensional signal features for content-based visual query. In Proceedings, *Symposium on Visual Communications and Image Processing*, May 1995. SPIE.

S.-F. Chang, J. R. Smith and J. Meng. Efficient techniques for feature-based image/video access and manipulation. In Proceedings, *33ʳᵈ Annual Clinic on Library Applications of Data Processing: Digital Image Access and Retrieval*, March, 1996.

C. Faloutsos, R. Barber, M. Flickner, W. Niblack, D. Petkovic and W. Equitz. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3, 1994.

J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner and W. Niblack. Efficient color histogram indexing for quadratic form distance functions. *IEEE Trans. Pattern Anal. Machine Intell.*, July, 1995.

K. S. Jones. *Information Retrieval Experiment.* Butterworth and Co., 1981

J. Meng and S.-F. Chang. Tools for compressed-domain video indexing and editing. In *Symposium on Electronic Imaging: Science and Technology – Storage & Retrieval for Image and Video Databases IV*, San Jose, CA, February 1996. IS&T/SPIE.

W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker and C. Faloutsos. The QBIC project: Querying images by content using color, texture and shape. In *Storage and Retrieval for Image and Video Databases*, vol. 1908, February 1993. IS&T/SPIE.

A. Pentland, R. W. Picard and S. Sclaroff. Photobook: Tools for content-based manipulation of image databases. In *Storage and Retrieval for Image and Video Databases II*, February 1994. IS&T/SPIE.

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*, San Jose, CA, February 1996. IS&T/SPIE.

J. R. Smith and S. F. Chang. Automated Binary Texture Feature Sets for Image Retrieval. In Proceedings, *Int. Conf. On Acoust. Speech and Sig. Proces. (ICASSP).* May, 1996. IEEE.

I. H. Witten, A. Moffat and T. C. Bell. *Managing Gigabytes: compressing and indexing documents and images.* Van Nostrand Reinhold, 1994.

D. Zhong, H. J. Zhang and S.-F. Chang. Clustering methods for video browsing and annotation. In *Symposium on Electronic Imaging: Science and Technology – Storage & Retrieval for Image and Video Databases IV*, San Jose, CA, February 1996. IS&T/SPIE.