Color and Texture based Image Retrieval

Harshada Anand Khutwad
Collage of Engg
Bharati Vidhyapeeth University Pune India

Ravindra Jinadatta Vaidya
DepartmentElectronics
H.O.D. of Electronics dep
Collage of Engg
Bharati Vidhyapeeth University Pune India

ABSTRACT
Content Based Image Retrieval is an interesting and most emerging field in the area of 'Image Search', finding similar images for the given query image from the image database. Current approaches include the use of color, texture and shape information. Considering these features in individual, most of the retrievals are poor in results and sometimes we are getting some non relevant images for the given query image. So, this dissertation proposes a method in which combination of color and texture features of the image is used to improve the retrieval results in terms of its accuracy. For color, color histogram based color correlogram technique and for texture wavelet decomposition technique is used. Color and texture based image retrieval computes image features automatically from a given query image and these are used to retrieve images from database

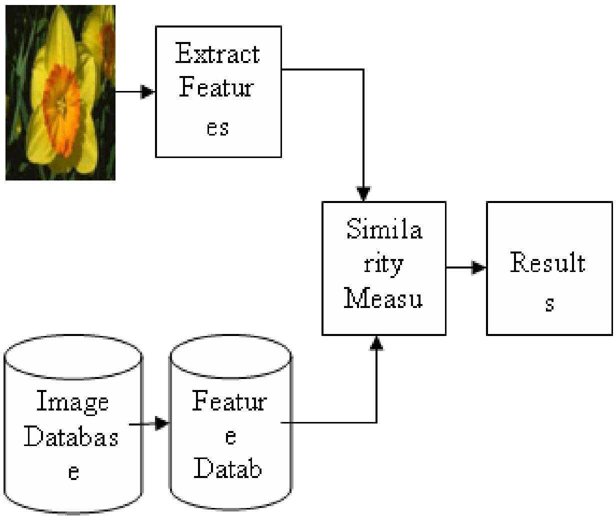
Key Words
CBIR; Color based Search; Texture based Searching; Color Histogram; Pyramid Structure Wavelet Transform model; Euclidean Distance; Quadratic Distance Metric.

1. INTRODUCTION
1.1 Motivation
Now a day, more and more attention is focused on content based image retrieval (CBIR) due to the tremendous growth of the number and sizes of digital image and video collections on web. So, it becomes necessary to develop power tools for retrieving this unconstrained imagery. In addition, CBIR is also the key technology for improving the interface between user and computer.

Most conventional image databases are text annotated .As a result, image retrieval is based on keyword searching. Text annotated images are simple and easy to manipulate. However, there are two major problems with this method. First, creating keywords for large number of images is time consuming. Moreover, the keywords are inherently subjective and not unique. Due to these disadvantages, automatic indexing and retrieval based on image content becomes more desirable for developing large volume image retrieval applications.

Research on multimedia systems and CBIR has been given tremendous importance during last decades. The reason behind this is the fact that multimedia databases deal with text, audio, video and image data which could provide us with enormous amount of information and which has affected our life style for the better. CBIR is a bottleneck of the access of multimedia databases simply because there is a vast difference in the perception capacity between a human and computer.

One of the primary challenges in the digital libraries is the problem of providing intelligent search mechanism for multimedia collections while there are good tools for searching text collections, images are much more difficult. If the images are annotated by hand, a textual search can be used, however, this approach is too laborious to scale up with large digital libraries. Automated methods for searching large database of images are therefore necessary. This in turn requires effective image features for comparing images based on their overall appearance.



1.2 Problem Definition
Color has been extensively used in image matching and retrieval. But color retrieval alone does not give good results. In this thesis we consider the texture of the image along with the color to improve the efficiency. Color and texture features are combined in our retrieval system to compute the similar images for the given query image from the database.

1.2 Scope
The scope of this research is circumscribed to CBIR system based on color and texture features of images to improve the efficiency of the CBIR system. We have computed the image features described in Chapter 6 on mixed natural database and implemented color and texture feature approach on that .Using color and texture feature, we retrieved images from the databases for the given query image.

1.3 Prior-Work
Early work on image retrieval can be traced back to the late 1970s. In 1979, a conference on Database Techniques for Pictorial Applications [6] was held in Florence. Since then, the application potential of image database management techniques has attracted the attention of researchers. Early techniques were not generally based on visual features but on the textual annotation of images. In other words, images were

first annotated with text and then searched using a text-based approach from traditional database management systems.

Text-based image retrieval uses traditional database techniques to manage images. Through text descriptions, images can be organized by topical or semantic hierarchies to facilitate easy navigation and browsing based on standard Boolean queries. However, since automatically generating descriptive texts for a wide spectrum of images is not feasible, most text-based image retrieval systems require manual annotation of images. Obviously, annotating images manually is a cumbersome and expensive task for large image databases, and is often subjective, context-sensitive and incomplete. As a result, it is difficult for the traditional text-based methods to support a variety of task-dependent queries.

In the early 1990s, as a result of advances in the Internet and new digital image sensor technologies, the volume of digital images produced by scientific, educational, medical, industrial, and other applications available to users increased dramatically. The difficulties faced by text-based retrieval became more and more severe. The efficient management of the rapidly expanding visual information became an urgent problem. This need formed the driving force behind the emergence of content-based image retrieval techniques. In 1992, the National Science Foundation of the United States organized a workshop on visual information management systems to identify new directions in image database management systems. It was widely recognized that a more efficient and intuitive way to represent and index visual information would be based on properties that are inherent in the images themselves. Researchers from the communities of computer vision, database management, human-computer interface, and information retrieval were attracted to this field. Since then, research on content-based image retrieval has developed rapidly [1]. Since 1997, the number of research publications on the techniques of visual information extraction, organization, indexing, user query and interaction, and database management has increased enormously. Similarly, a large number of academic and commercial retrieval systems have been developed by universities, government organizations, companies, and hospitals.

2. CONTENT COMPARISON TECHNIQUES

There are some common methods for extracting content from images so that they can be easily compared. The methods outlined are not specific to any particular application domain.

2.1 Color Retreival

Color is the most extensively used visual content for image retrieval. Its three dimensional values make its discrimination potentiality superior to the single dimensional gray values of images. Before selecting an appropriate color description, color space must be determined first. Retrieving images based on color similarity is achieved by computing a color histogram for each image that identifies the proportion of pixels within an image holding specific values. The first order (mean), the second order (variance) and the third order (skew ness) color moments have been proved to be efficient and effective in representing color distributions of images.

A different way of incorporating spatial information into the color histogram, color coherence vectors (CCV), was proposed. Each histogram bin is partitioned into two types, i.e., coherent, if it belongs to a large uniformly-colored region, or incoherent, if it does not. Another method called color correlogram expresses how the spatial correlation of pairs of colors changes with distance.

2.2 Texture Retrieval

Texture is a widely used and intuitively obvious but has no precise definition due to its wide variability. Visual texture in most cases is defined as a repetitive arrangement of some basic pattern. This repetition may not be random. However, a texture pattern normally has some degree of randomness due to randomness in basic pattern as well as due to randomness in the repetition of basic pattern. To quantify texture, this randomness is measured by some means over a small rectangular region called window.Thus, texture in an image turns out to be a local property and depends on the shape and size of the window .Identifying a patch in an image as having uniform texture or discriminating different visual textures obeys the law of similarity. In this case, the texture property is used to produce similarity groupings.

Basically, texture representation methods can be classified into two categories: *structural* and *statistical*. Structural methods, including *morphological operator* and *adjacency graph*, describe texture by identifying structural primitives and their placement rules. They tend to be most effective when applied to textures that are very regular. Statistical methods, including Fourier power spectra, co-occurrence matrices, shift-invariant principal component analysis (SPCA), Tamura feature, Wold decomposition, Markov random field, fractal model, and multi-resolution filtering techniques such as Gabor and wavelet transform, characterize texture by the statistical distribution of the image intensity

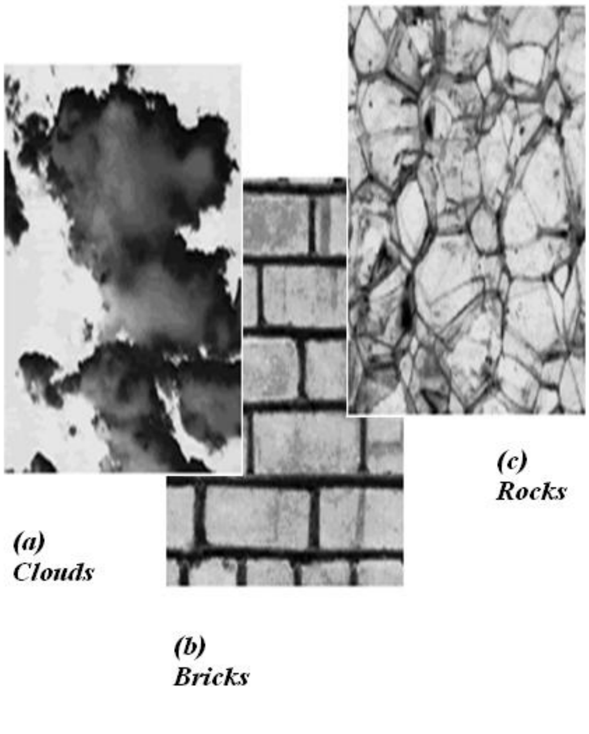


Figure 2.2: Different types of texture

2.3 Proposed Scheme for CBIR Using Color and Texture

The management of a large number of images in a multimedia database has received much attention in recent years. Most of the earlier works are largely focused on techniques to extract useful information (such as color, texture or shape) that represents images. Rapid retrieval is becoming important issue as image databases continue to grow in size and a slow will no longer be acceptable to the user community.

In this paper we present retrieval system that supports color and texture features. In our system image is represented by set of features that are extracted by using color and texture properties of image which are described in this chapter. These extracted features are then compared with the stored images features in the database. The retrieved images are ranked and top results are then displayed.

3. COLOR FEATURE

Color histogram is widely used for color based image retrieval in content-based image retrieval (CBIR). A color histogram describes the global color distribution of an image. It is very easy to compute and is insensitive to small changes in viewing positions. However, the histogram is not robust to large appearance changes and it also gives different results as colors are same in the images. On the other hand, color Correlogram is efficiently used for image indexing in content-based image retrieval. Color Correlogram extracts not only the color distribution of pixels in images like color histogram, but also extracts the spatial information of pixels in the images. So, we use the image bin separation technique followed by extracting maxima of frequencies and constructing its Correlogram.

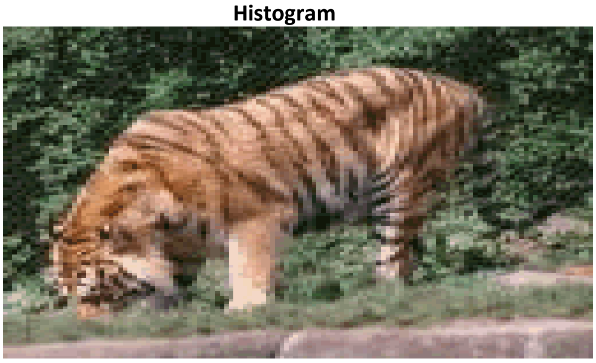
3.1 Color algorithm

Calculate the histogram of an image.

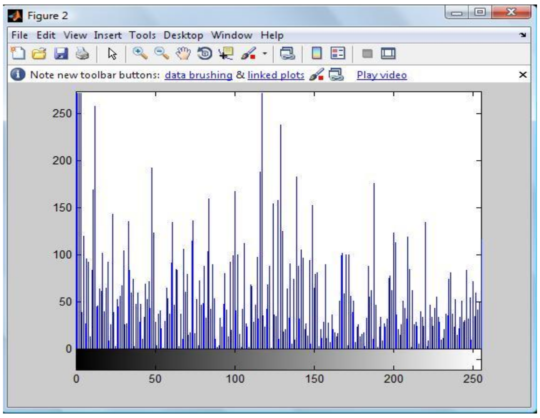
- After that calculate the flat histogram of an image i.e. equalized histogram.
- Divide the calculated histogram into four equal sized bins.
- Further sub-divide each calculated equal sized bins into four more bins.
- Calculate the correlogram and construct it.
- Calculate the distance between constructed correlogram of query image and database image.

As we all know that RGB and indexed images carry high values that require more computation time. In this project we have used 8-bit images and hence values lie in the range 0-255. Therefore reduction in the computation time and power required for extracting features from an image. The resulting image undergoes histogram equalization in order to enhance contrast of values of an image by generating its flat histogram.

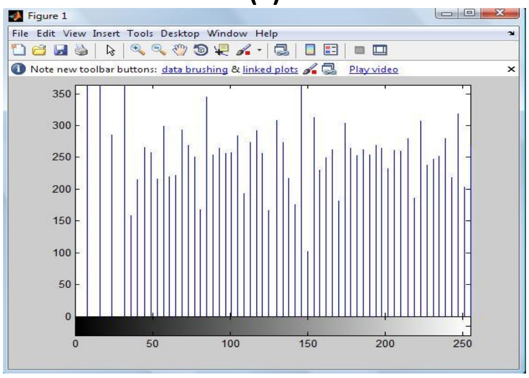
Histogram



(i)



(ii)



(iii)

Figure 3.1 :(i)Image (ii)Its Histogram(iii)Its Equalized

The histogram equalized image is split into four fixed bins in order to extract more distinct information from it. The frequencies of 256 values of images are split into four bins carrying 64 values each (0~63, 64~127, 128~191, and 192~255). This is done by turning off the gray values of image which do not lie between the four bins.

This provides a better illustration of image segments and simplifies the computation of features for the distinct portion of image. The bins produced are further subdivided into four fixed ranges carrying 16 values each. For example, for the first bin carrying value range of 0 to 63, four sub-divisions of the frequency values will be 0~15, 16~31, 32~ 47, and 48~63. Similar procedure is repeated for rest of the bins.

The information from bins and sub-divisions are stored in the form of a Correlogram as shown below

Figure 3.2: Correlogram of Bins Vs Subdivision

Sub-Division	1	2	3	4
Bin1	1024	455	567	543
Bin2	789	888	390	234
Bin3	557	459	890	567
Bin4	689	890	598	897

3.2 Similarity Measurement for Color

- Calculate the flat histogram of image.
- Get the correlogram of the query image.
- For every image i in the database obtain the correlogram.
- Calculate the Euclidean distance between the two sets of correlogram that is between query image and image i in the database.
- Increment ***i***. Repeat from step 3.

The distance measurement process comprises of three steps. The Correlogram matrices are subtracted, at first, under simple subtraction rules. Secondly, the sum of the matrix components is calculated. Finally, the third step comprises of sorting the absolute values of the sums obtained in the second. For Example for two images A and B let their corresponding correlograms be 4 row and 4 column matrices. Let, Correlogram of image A be Corr(A).Similarly, let the Correlogram of image B be Corr(B*)*. According to Euclidean distance algorithm described above, let:

Distance (A, B) = Corr(A)-Corr(B)

Where, Distance (A, B*)* is a vector containing the result of difference calculation as described in equation. The components of the resulting matrix are summed together and the absolute value of this sum is considered

4. TEXTURE FEATURE

We used a method called the pyramid-structured wavelet transform for texture classification. Its name comes from the fact that it recursively decomposes sub signals in the low frequency channels. It is mostly significant for textures with dominant frequency channels. For this reason, it is mostly suitable for signals consisting of components with information concentrated in lower frequency channels [2]. Due to the innate image properties that allows for most information to exist in lower sub-bands, the pyramid-structured wavelet transform is highly sufficient.

Using the pyramid-structured wavelet transform, the texture image is decomposed into four sub images, in low-low, low-high, high-low and high-high sub-bands. At this point, the energy level of each sub-band is calculated .This is first level decomposition. Using the low-low sub-band for further decomposition, we reached fifth level decomposition, for our project. The reason for this is the basic assumption that the energy of an image is concentrated in the low-low band. For this reason the wavelet function used is the Daubechies wavelet.

4.1 Texture algorithm

1. Decompose the image into *four* sub-images

2. Calculate the energy of all decomposed images at the same scale, using [2]:

$$E = \frac{1}{MN} \sum_{i=1}^{m} \sum_{j=1}^{n} \left| X(i, j) \right|$$

Where ***M*** and ***N*** are the dimensions of the image, and ***X*** is the intensity of the pixel located at row ***i*** and column ***j*** in the image map.

3. Repeat from step 1 for the low-low sub-band image, until index **ind** is equal to 5. Increment **ind.**

Using the above algorithm, the energy levels of the sub-bands were calculated and further decomposition of the low-low sub-band image. This is repeated five times, to reach fifth level decomposition. These energy level values are stored to be used in the Euclidean distance algorithm.

4.2 Similarity measurement for texture

We have used Euclidean Distance Algorithm for comparing distance between texture features of query image and database images .And the algorithm is as follows

- Decompose query image.
- Get the energies of the first dominant ***k*** channels.
- For image ***i*** in the database obtain the ***k*** energies.
- Calculate the Euclidean distance between the two sets of energies, using [2]:

$$D_i = \sum_{k=1}^{k} \left(x_k - y_{i,k} \right)^2$$

- Increment ***i***. Repeat from step 3.

Using the above algorithm, the query image is searched for in the image database. The Euclidean distance is calculated between the query image and every image in the database. This process is repeated until all the images in the database have been compared with the query image. Upon completion of the Euclidean distance algorithm, we have an array of Euclidean distances, which is then sorted and top results are then displayed.

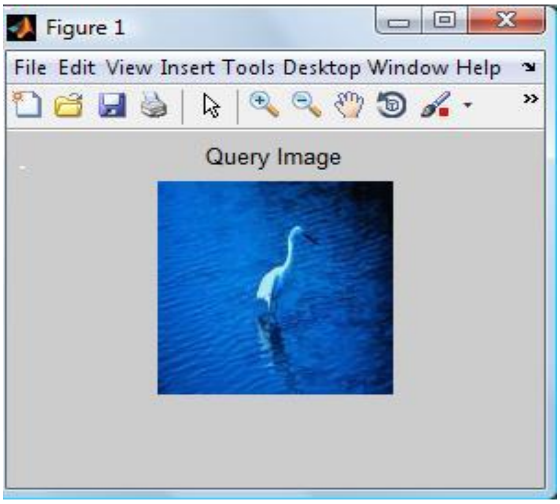
5. DEVLOPMENT ENVIRONMENT

We have developed our application in MATLAB which is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows us to solve many technical computing problems, especially those with matrix and vector formulations. It takes a fraction of the time to write a program in a scalar non interactive language such as C and FORTRAN. MATLAB features a family of add-on application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow us to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

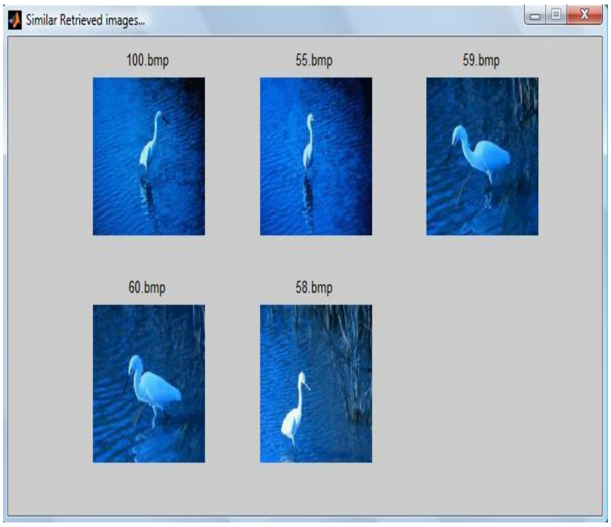
The MATLAB system consists of five main parts:

- Development Environment
- The MATLAB Mathematical Function Library
- The MATLAB Language
- Graphics
- The MATLAB Application Program Interface (API

Obtained Result

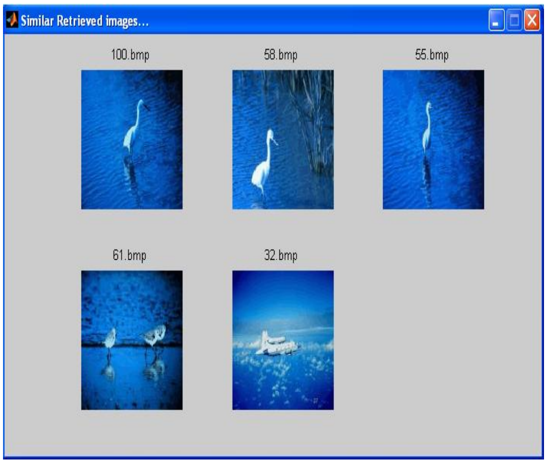


(i)



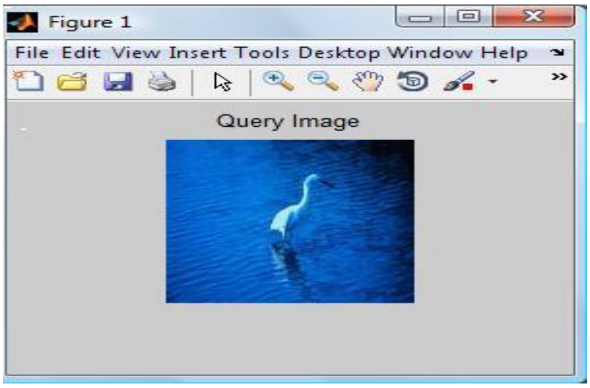
(ii)

Figure 4.5 :(i) Query image (ii) Top five retrieved images using color feature only



(ii)

Figure4.6: (i) Query Image (ii) Top five retrieved images using color and texture



(i)

6. CONCLUSION

The dramatic rise in the sizes of images databases has stirred the development of effective and efficient retrieval systems. The development of these systems started with retrieving images using textual connotations but later introduced image retrieval based on content. This came to be known as CBIR or Content Based Image Retrieval. Systems using CBIR retrieve images based on visual features such as color, texture and shape, as opposed to depending on image descriptions or textual indexing. In this project, we have researched various modes of representing and retrieving the image properties of color, texture and shape. The application performs a simple color-based search in an image database for an input query image, using color histogram based correlogram. It then compares the color correlogram of different images using the Euclidean Distance Equation. Further enhancing the search, the application performs a texture-based search in the color results, using wavelet decomposition and energy level calculation. It then compares the texture features obtained using the Euclidean Distance Equation. And it is shown that our method gives better results than taking color only.

7. FUTURE WORK

The results Of CBIR system presented in this thesis are based on color and texture features, meaning that a simple color based search is done first followed by texture and these two combined to form feature vector and are used for image retrieval. The results may be improved if we consider shape along with color and texture and we can also use some indexing scheme for better retrieved results and performance.

8. ACKNOWLEDGEMENTS

Our thanks to the Prof. Mr. Sakhare sir & Prof.Mr. Vaidya sirwho have contribute towards development.

9. REFERENCES

[1] Fuhui Long, Hongijang Zhang, David Dagan Feng, "Fundamentals of Content Based Image Retrieval". http://research.microsoft.com/asia/dload_files/group/mcomputing/2003P/ch01_Long_ v40-proof.pdf

[2] Sharmin Siddique, "A Wavelet Based Technique for Analysis and Classification of Texture Images," Carleton University, Ottawa, Canada, Project Report 70.593, April 2002.

[3] Chin Chen-Chaur, Ting Chu Hsueh, "Similarity Measurement between Images", IEEE Annual International Computer Software and Applications Conference (COMPSAC'05), pp.207-248, 2005.

[4] Timothy K.Shih,Jung-yao Huang,Ching-Sheng Wang,Jason C.Hung And Chuan-Ho Kao, " An intelligent content-Based image retrieval System based on color ,Shape and spatial relations" Proceedings Natl.Si.Counc.Roc(A), volume 25,No.4, pp.231-243,2001.

[5] Virginia E. Ogle and Michael Stonebraker, "Chabot: Retrieval from a relational database of images", IEEE Computer, 28(9), pp.40–48, September 1995.

[6] A. Blaser, "Database Techniques for Pictorial Applications", Lecture Notes in Computer Science, Springer Verlag GmbH, Volume 81,1979.

[7] J. Kreyss, M. R¨oper, P. Alshuth, Th. Hermes, and O. Herzog, "Video retrieval by still image analysis ImageMiner", In Proceedings of IS&T/SPIE's Symposium on Electronic Imaging: Science & Technologies, pp.8-14 Feb. '97, San Jose, CA, 1997.

[8] S. Sclaroff, L. Taycher, and M. La Cascia, " Imagerover: A content-based image browser for the world wide web", In Proceedings IEEE Workshop on Content-based Access of Image and Video Libraries, June '97, 1997.

[9] Leonid Taycher, Marco La Cascia, and Stan Sclaroff, "Image digestion and relevancefeedback in the ImageRover WWW search engine", In Proceedings of the 2nd International Conference on Visual Information Systems, San Diego, December '97, pp 85–94, 1997.

[10] M. La Cascia and E. Ardizzone, "Jacob: Just a content-based query system for video databases", In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP96),May 7-10, '96, Atlanta, Georgia, 1996.