Proceddings of Multimedia Modeling ’96 (MMM’96),
JP Courtiat, M Diaz & P. Senac (Eds), Toulouse, France, 1996, 12-15.

USING DOMAIN KNOWLEDGE IN QUERYING IMAGE DATABASES

TAT-SENG CHUA, KAH-CHOO TEO, BENG-CHIN OOI and KIAN-LEE TAN
Department of Information Systems and Computer Science
National University of Singapore

Current image retrieval systems use associated text and visual attributes of images, such as colors, textures or shapes, as indices for retrieval. The use of these attributes has been quite successful in retrieving a large proportion of relevant images. However, as each attribute is able to capture only part of the images’ semantics, retrieval techniques based on the combination of these attributes could only achieve limited retrieval effectiveness. In order to exploit the strong points of each attribute in identifying relevant images, we investigate the use of domain knowledge to guide the retrieval process. The retrieval model attempts to model the human's mental process of filtering and identifying dominant attributes in relevant images. In order to make the system domain independent, the domain knowledge is considered to be an expert knowledge supplied by the users as part of the query. A concept specification language is designed for this purpose. We test the system on a large image collection of over 12,000 images. The results of testing demonstrate that the use of domain knowledge could lead to substantial improvement in retrieval effectiveness.

Keywords: knowledge-based query; content-based image retrieval; domain knowledge.

1. Introduction

The advances in computer technology enable us to store and manipulate a massive amount of color images on the computer. Databases containing large on-line image collections are becoming common. Current and potential applications for imaging and other media include: medicine; photo-journalism; marketing and promotion; education and training; and entertainment. The emergence of large and complex image databases has led to the demand for tools that can manage, organize and retrieve images effectively.

Most early image retrieval systems use keywords or free-text descriptions supplied by the authors as the basis for retrieval [1, 14]. The problem with text-based retrieval systems is that the text descriptions must be entered manually by the authors, and hence the descriptions tend to be incomplete and inconsistent. Because of this problem, recent systems [5, 6, 8] use the visual attributes of images, such as colors, textures, and shapes, as indices for retrieval. The visual attributes can often be extracted directly from the images' contents. Thus, the indexing and retrieval of images using these attributes can be fully automated. However, the visual attributes are only good at describing objects with concrete visual forms, but not abstract concepts. Thus, the retrieval results are not very precise. Retrieval system that combines the use of colors and textures could achieve a retrieval effectiveness of up to 60% in recall and precision [5].

It is interesting to note that different retrieval models based on different combination of visual attributes and text descriptions achieve almost similar levels of retrieval effectiveness. Moreover, each model is able to retrieve a different subset of relevant images. One of the reasons is that each image attribute is only able to capture part of an image's semantics. Thus, if we are able to exploit the strong points of each attribute in identifying relevant images, the retrieval effectiveness of the combined system can be improved. A similar problem is encountered in free-text retrieval systems using different retrieval models [2].

This paper investigates the use of domain knowledge of the image collection to guide the selection of appropriate attributes for image retrieval. The knowledge-based model is based loosely on human's mental process of filtering and discriminating dominant attributes in relevant images. The resulting model is a hierarchy of real-world objects together with clues for detecting these objects in terms of colors, textures and text descriptions at the lowest level.

This paper describes the design and implementation of an image retrieval system that utilizes domain knowledge to retrieve images. In order to make the system domain independent, the required domain knowledge is not pre-defined by the system; rather, it is considered to be an expert knowledge that the users will supply based on their information needs. A concept specification language has been designed for this purpose. The paper discusses the testing of the system on a large image collection containing over 12,000 images.

2. Reviews of Related Works

Many existing image retrieval systems use keywords or free-text descriptions supplied by the authors as the basis for retrieval [1]. These systems mainly employ best-match information retrieval (IR) techniques [15] to retrieve a ranked list of relevant images based on users' free-text queries. The effectiveness of a text-based system is only as good as the text descriptions, which tend to be incomplete and inconsistent. Furthermore, the original text descriptions often do not allow for unanticipated search in subsequent retrievals.

One natural way to overcome some of the problems of text-based systems is to use the visual attributes of images for retrieval. The visual attributes used include colors, textures and shapes. Some systems employ heuristic shape measures, such the circularity, eccentricity, major axis orientation, and algebraic moment of invariant etc., to model dominant shape attributes [13]. Practical content-based image retrieval systems tend to use a combination of colors, textures and heuristic shapes as these attributes can be extracted automatically from the images' contents. In these systems, users formulate their queries either by supplying a sample image, or directly specifying a combination of visual attributes. The use of exact shape in retrieval is less successful and is confined mainly to specialized domain because there is a need for a pre-defined database describing the shapes to be matched [6]. [3] considered only the color contents of an image and employed the fuzzy set theory to express the intrinsic uncertainty of human in evaluating color similarities. [5] studied the use of combined query involving both colors and textures in a similarity-based retrieval that also takes into account of perceptually similar colors. Results from [5] indicated that the use of combined query could lead to substantial improvements in retrieval effectiveness. [8] considered the use of colors, textures, shapes and sketches for retrieval.

In an attempt to combine colors and object topology in retrieval, [12] used color pairs to model the distinct boundaries and relationships between objects in an image. In order to eliminate the contributions of the image background and to perform object-level retrieval, [6] considered the use of segmented images in retrieval. Coupled with the use of perceptually similar colors, substantial improvements in retrieval effectiveness were obtained. [10] developed a color-spatial model, which encodes image contents as a set of dominant colors, along with their spatial distributions throughout the image.

Text descriptions and visual attributes tend to be ambiguous in meanings, and hence they are difficult to be used to express precise query. The problem is aggravated as users' information needs are usually imprecise. The combination of these factors partly contributed to the low retrieval effectiveness of image retrieval systems. This problem has long been recognized in the IR community. Two main approaches have been used to overcome this problem. One approach relies on user's relevance judgments of earlier retrievals to enhance subsequent retrieval effectiveness. The relevance judgment information has been used effectively to re-formulate the users' queries and/or extend the images' text descriptions automatically [14]. The relevance judgment information can also be used explicitly by the users to specify a new query or to browsing the image database directly. The second approach considers the judicious use of domain knowledge to achieve higher retrieval effectiveness. Examples of such systems include RUBRIC [18] and I^3R [7] for free text, and a prototype system [19] for multimedia data. In these systems, domain knowledge is modeled as a collection of concepts. Each concept contains the descriptions, the relationships with other concepts, and the rules for its recognition. In I^3R and the multimedia prototype [19], the knowledge base is pre-defined by the authors and is used to assist the users in formulating queries for more accurate retrieval. RUBRIC takes a different approach of requiring the users to define the concept structure as part of the queries.

3. A Model for Knowledge-Based Image Retrieval

Image has rich contents from which we can derive many concepts. The full semantics of an image are manifested through multiple visual and non-visual forms. The use of just text or a visual attribute alone is unable to fully capture the image's semantics as each attribute has limited expressive power on its own. For example, colors and textures are good at capturing the visual aspects of an object with concrete form. However, they are not very precise and are incapable of representing abstract concepts presence in images. Text descriptions, on the other hand, are good at describing abstract concepts. However, they are weak in describing visual contents, and tend to be incomplete and inconsistent. Thus, each attribute tends to capture different facets of the images' contents. Because of these problems, retrieval models based on different content attributes tend to retrieve different subset of relevant images, and with limited retrieval effectiveness.

To overcome the representation problems of each attribute, all attributes should be used together to provide a more complete description of the images' contents. This approach has been taken by many researchers in utilizing multiple attributes for image retrieval [5, 8, 9]. All approaches use heuristic ratios to combine evidence arriving from each attribute. The ratios are determined experimentally. The use of multiple attributes has led to improvement in retrieval effectiveness [5]. However, it is not very satisfactory as different attributes tend to have different degree of importance for different classes of queries.

One way to tackle this problem is to use domain knowledge of the image collections and queries to judiciously combine evidence of different attributes. The use of domain knowledge is based loosely on the human's mental model in filtering and discriminating suitable attributes when identifying certain types of images. One typically starts with a stereotyped model of the concept that one is looking for, be it a tree or a park. There could be multiple models of the same concept. The stereotyped model contains information that can be used to identify the concept in the image in terms of other sub-concepts and image content attributes. This is similar to the idea of frames as discussed in [11]. The degree of similarity between the contents of an image and the concept model is used to deduce the relevance of the image.

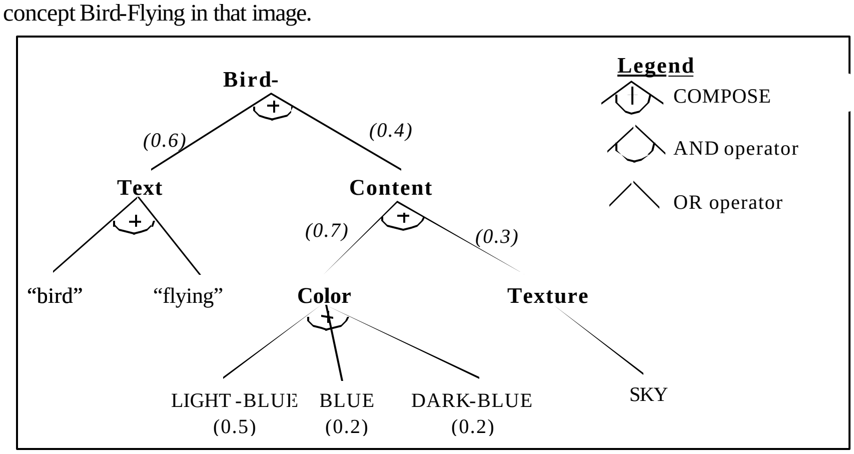
For example, if one is looking for images of trees, one would look for, in the order of importance, certain specific textures and colors that characterize a tree, together with typical text terms used to describe the tree. Shape might not be an important attribute here. At the higher level, if one is looking for images of a park, one may start with a mental image of a large green field consisting of trees and possibly some flowers and a pond. Thus the stereotyped model of the concept park is composed of a number of primitive sub-concepts. Each primitive concept is in turn defined in terms of a set of image content attributes. With this model, we can successively build up a hierarchical model of real-world concepts together with clues for their detection in terms of colors, textures and text descriptions at the primitive level.

4. Modeling of Concept-Based Query for Image Retrieval

Most knowledge-based systems attempt to maintain a knowledge base that encodes all domain knowledge gathered from various domain experts. An image has rich contents that many different concepts may be derived from it. It is impossible and impractical to attempt to define a common knowledge base that can be shared by all users. Our approach is to provide a mechanism for the users to incorporate domain knowledge into their queries. This is the approach taken in RUBRIC [18] for free-text retrieval. This approach not only overcomes the problem of incomplete knowledge inherent in most knowledge-based systems, but also permits the users to structure their queries based on their perceptions of a concept.

Similar to RUBRIC, we model the query as a hierarchy of concepts, except that in our case the leaf concepts are defined in terms of multiple images' content attributes including text, colors and textures. Each concept has three components: its name; its relationships with other concepts; and in the case of primitive leaf concepts, rules for its identification within the images' contents.

Figure 1 shows the hierarchy for the primitive concept Bird-Flying. Here the concept Bird-Flying is defined in terms of two attribute, text and content, with the content being further defined in terms of color and texture attributes. At the primitive leaf level of the concept hierarchy, the text attribute is defined as a combination of text strings "“bird” and “flying”; the color attribute is defined as a composition of LIGHT-BLUE, BLUE, and DARK-BLUE colors; while the texture attribute is defined to be of type SKY. The values of text, color and texture attributes are used to match against the content of an actual image to deduce the presence of concept Bird-Flying in that image.



Note: Bold words denote the names of attributes or concepts; capital words and words enclosed in quotes give the values of attributes.

Figure 1: Concept tree of a primitive concept Bird-Flying

In a concept hierarchy, operators must be defined to describe how lower level concepts or attributes may be combined to form higher level concepts. Consider first the combination of attribute values to derive the value of an image attribute. Three operators are defined for this purpose; they are the AND, OR and COMPOSE operators. The AND and OR are the usual operators that derive the meaning of an attribute based on the conjunction and disjunction of the specified values respectively. These operators are normally used in boolean "exact match" environment. In an image retrieval environment characterized by unstructured information and imprecise query specification, the ability to express similarity-based matching is also essential. Similarity-based matching denotes the use of best match, rather than exact match, to derive the degree of similarity between the content of an image and the query. To express similarity-based matching, we introduce an additional operator, called COMPOSE. For example, the COMPOSE operator is used in Figure 1 to define the composition of color attribute in terms of 50% LIGHT-BLUE, 20% BLUE, and 20% DARK-BLUE. It means that relevant images should preferably contain the desired color composition. It does not require all three colors to be present in the relevant images as for the AND operator. It is also insufficient for a relevant image to contain just 20% of BLUE as expressed by the OR operator.

The same three operators are also used at the higher level to combine lower level concepts to form high-level concepts. Here the COMPOSE operator denotes that the meaning of the higher level concept is derived by taking the weighted evidence of the lower level concepts. In the Bird-Flying concept shown in Figure 1, the evidential value of Bird-Flying is derived by combining the weighted evidence of the text and content attributes.

User may attach weight to an attribute or concept to denote its importance in inferring the higher-level concept. With reference to the Bird-Flying concept, weights of 0.6 and 0.4 are attached to the text and content attributes respectively. The weight denotes the degree of importance judged by the human user, based upon his or her experience and insight; it is not a statistical quantity. The weights will influence the combination of attributes and ultimately the ranking of the images retrieved. The weight is a real number in the interval [0,1]. If the weight is omitted for an attribute, it is assumed to be 1.0 (i.e. absolute importance). To ensure closure property when applying the operator, the sum of all weights associated with an operator should be less than or equal to 1.0.

To facilitate the definition of concept-based query by the users, a concept specification language has been defined. The specification of concept Bird-Flying is given in Figure 2. In this language, we use the symbols +, & and | to denote the operators COMPOSE, AND and OR respectively. The ATTRIBUTE field of a concept is used to define its sub-concepts to form the concept hierarchy. The rest of concept specification should be self explanatory. Further details on the design and syntax of the concept specification language can be found in [17].

(**CONCEPT** Bird-Flying)
(**DESCRIPTION** Bird-Fly ing "Birds of any color flying")
(**ATTRIBUTE** Bird-Flying ((**TEXT** 0.60) + (**CONTENT** 0.40)))
(**TEXT** Text "bird" + "flying")
(**CONTENT** Bird-Flying Content)
(**ATTRIBUTE** Content ((**COLOR** 0.70) + (**TEXTURE** 0.30)))
(**COLOR** Content ((LIGHT-BLUE 0.5) + (BLUE 0.2) + (D ARK-BLUE 0.2)))
(**TEXTURE** Content SKY)

Figure 2: The specification of concept Bird-Flying

We can use the definition of primitive concepts to construct higher level concepts. An example of a higher-level concept is the scene of birds flying during the sunset. This concept, Bird-Flying-in-Sunset, is built upon two primitive concepts, namely Bird-Flying and Sunset. Figure 3 shows its concept tree. In a similar manner, queries expressing multi-level concepts can also be constructed.

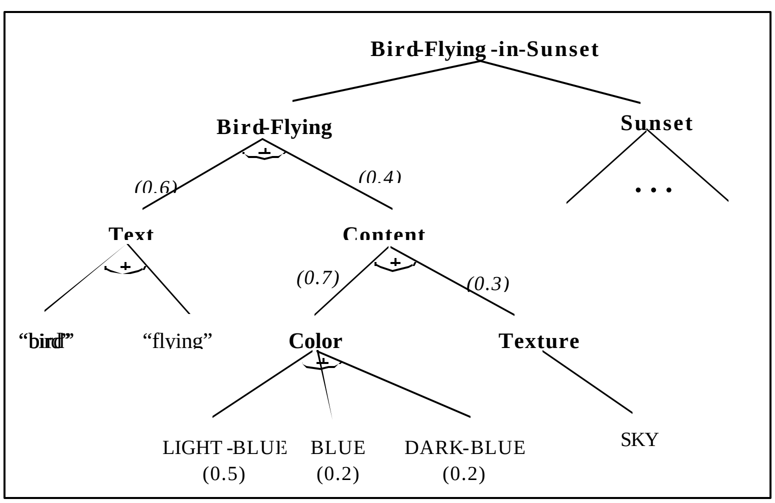


Figure 3: Concept tree of a two-level concept

5. Query Processing

The concept-based query entered will be verified by the system to ensure that it is syntactically correct and that all the weights and attribute values specified are valid. If no error is detected, the system will proceed to construct the concept tree starting from the root concept. Examples of that concept trees maintained by the system are shown in Figures 1 and 3. Based on the contents of each image, a similarity value is computed for the root concept of the concept tree. This value is used to denote the similarity between the image and the concept-based query.

The computation of similarity between the root concept and the image starts from its leaf nodes where a set of image content attributes is specified. For each attribute value (such as SKY) or a group of attribute values linked together by the COMPOSE operator (such as the text strings “bird” and “flying”), we first compute its similarity with the content of the image. The similarity is computed using an appropriate similarity formula depending on the type of attribute. For text attribute, we employ the vector space IR model and the cosine similarity formula [15]. For colors, we employ a variant of histogram-based similarity formula that takes into account of perceptually similar colors [6, 17]. The color computation is carried out in the CIE L*u*v color space [4]. For texture attribute, we extend the texture measures developed in [16] for similarity computation [17].

The similarity values computed at the leaf level are propagated up the concept hierarchy. There are three ways of propagating the similarity values -- depending on the types of operators used to combine the lower-level concepts or attributes. For the *OR* operator, the combined similarity value is derived simply by taking the maximum similarity value of its child nodes. The resulting function is expressed as:

$$SIM_{OR} = \mathrm{MAX}\,\{SIM_1 , SIM_2 , ..., SIM_N\} \qquad (1)$$

where SIM_i is the similarity value contributed by its ith child node.

The combined similarity value of an *AND* operator is derived by taking the minimum of the similarity values propagated by its child nodes. It is defined as:

$$SIM_{AND} = \mathrm{MIN}\,\{SIM_1 , SIM_2 , ..., SIM_N\} \qquad (2)$$

Finally, the similarity for a *COMPOSE* operator is derived by taking the vector sum of the similarity values propagated by its child nodes. Let $W_1, W_2, ..., W_N$ represent the weights assigned to the respective child node, the combined function is expressed as:

$$SIM_{COMPOSE} = \sum_{i=1}^{N}(W_i * SIM_i) \tag{3}$$

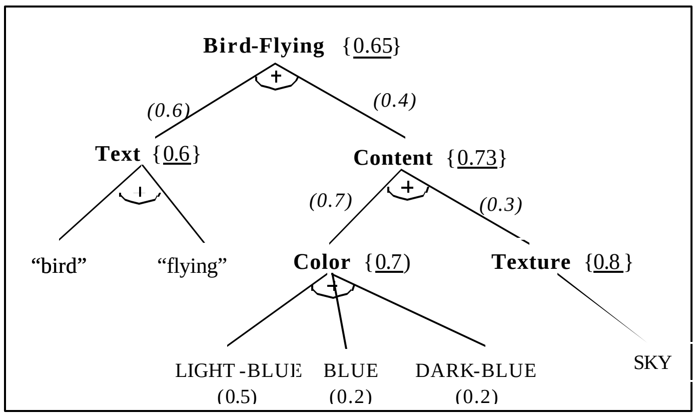


Figure 4: Computation of Similarity Value for an Image

Figure 4 illustrates how similarity values are propagated for the hierarchy of concept Bird-Flying. Note that the arcs without the weight specifications are assumed to have the weights of 1.0. For clarity, similarity values are underlined and enclosed by braces. Assuming that for a given image, the similarity values of 0.6, 0.7 and 0.8 are obtained for text, color and texture attributes respectively. Figure 4 shows clearly the process of propagating these weights up the concept hierarchy. The final weight obtained for the root concept is 0.65. This value denotes the degree of similarity between the concept-based query and the specific image.

6. System Design and Implementation

The overall design of the system is given in Figure 5. At the heart of the system is the knowledge-base retriever, which performs the task of analyzing the users' concept-based queries and retrieving the images. It activates the appropriate attribute (text, color and texture) retrievers to perfom the content level matching.

The image database holds the set of 12,000 images used for testing. The images, together with the associated text descriptions, are acquired commercially from Kagema Corporation. The images are divided into eighteen categories by the image supplier. Major categories include:

art, computer, food, nature, travel, animal, etc. In order to speed up query processing, images are pre-processed to extract the appropriate attribute representations. These attribute representations are stored in different index files. The inverted index file is used for text, while signature files are used for colors and different texture measures.

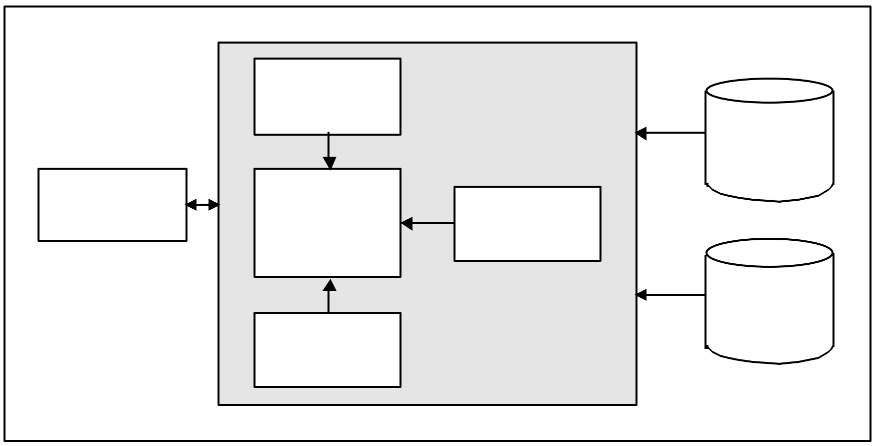


Figure 5: Overall design of the system

The graphical interface is designed to display the top 120 retrieved images ranked in decreasing order of similarity. The interface supports the input of different types of queries. A textual interface is provided to facilitate the users in entering, editing, viewing and executing the concept-based queries. Users are also able to specify content-based queries by: (a) directly choosing from the palette of colors and texture patterns, and/or entering the text descriptions; and (b) selecting a sample image.

The system is implemented on the SiliconGraphcis Indigo workstation using C++ programming language. The user interface is written in X/Motif.

7. Testing and Evaluation

We test the system on an image database containing about 12,000 images, covering a wide range of categories. As it is impossible to design queries that test all categories of images, we design our queries based only on two categories: animals and nature. These two categories are chosen arbitrarily. Note however that the retrieval process does not restrict to searching for images only in these two categories; all 12,000 images of the database are searched.

A total of eight queries are designed and they are summarized in the table 1. Each query is based on a sample image taken from the database. For each query, a concept-based query is formulated. The concept-based specifications of queries 1 and 2 are given in Figures 6 and 2 respectively. A sample screen showing the top eighteen images retrieved using query 1 is shown in Figure 7b.

For each query, we pre-determine a set of relevant images. The relevant images are obtained manually by a combination of retrieval and exhaustive browsing. Typically, we start the search by issuing the query involving text and color attributes to get an initial image set. This set is then refined and expanded by using query by sample image and exhaustive browsing The estimated number of relevant images for each query is also shown in Table 1. Please note that the number of relevant images for each query is only an estimate as it is difficult to get the exact figure for a database of over 12,000 images.

To demonstrate the effectiveness of the concept-based queries, the result of the knowledge-based system is compared with a system that do not utilize domain knowledge. The non-knowledge-based system is similar to the one developed in [5, 8]. It uses multiply content attributes for retrieval. The content attributes used are text, colors and texture. A fixed set of ratios is used to combine the similarity values derived from each content attribute. The ratios are determined through exhaustive experiments to provide the best retrieval effectiveness. The sample image characterizing each query is used as input query to the non-knowledge-based system.

Domain	Query Id	Query Name	Query Description	# of relevant images
Animals	Q1	Ape	Monkey, Orang-utan and gorilla	40
	Q2	Bird Flying	Flying birds of any color	40
	Q3	Red or White Bird	Red or white birds only	50
	Q4	Horse	Horse in outdoor environment	50
Nature	Q5	Pink Flower	Flower in any shade of pink color	40
	Q6	Sunset	Scenes of sunset	50
	Q7	Scenery	Scenes of mountain, sea, river and sky	50
	Q8	Red or Yellow Flower	Images of red or yellow flowers only	70

Table 1: Summary of queries used for testing

(**CONCEPT** Ape)
(**DESCRIPTION** Ape “Monkey, Orang-utan and Gorilla”)

(**ATTRIBUTE** Ape ((**TEXT** 0.60) + (**CONTENT** 0.40)))
(**TEXT** Text (“monkey” | “gorilla” | “orang-utan”))
(**CONTENT** Ape Image-Ape)
(**ATTRIBUTE** Image-Ape ((**COLOR** 0.60) + (**TEXTURE** 0.4)))
(**COLOR** Image-Ape (DARK-BROWN 0.10))
(**SAMPLETEXTURE** Ape-Image “~/animals/1281006a.jpg”)

Figure 6: Concept-based specification of query 1 (Ape)

Since it is not possible to retrieve all relevant images in an image collection of over 12,000 images, our experiment evaluates only the top 60 images ranked by the system. Since not all relevant images are retrieved, the commonly used evaluation measures such as Recall-Precision tables or graphs are not appropriate. Instead, we compute the normalized precision (P_{norm}) and normalized recall (R_{norm}) [15] for each query. This is done both for our system (KB system) and the system that does not utilize domain knowledge (non-KB system). Table 2 summarizes the results of both systems. Table 2 also gives the percentage improvements in normalized precision and recall for the KB system over the non-KB system.

	KB System		Non-KB System		Change (%)	
Query Id	**P_{norm}**	**R_{norm}**	**P_{norm}**	**R_{norm}**	**P_{norm}**	**R_{norm}**
Q1	0.84	0.91	0.27	0.52	211	76
Q2	0.68	0.77	0.61	0.74	11	5
Q3	0.59	0.74	0.36	0.57	66	31
Q4	0.83	0.90	0.78	0.85	7	6
Q5	0.83	0.90	0.76	0.85	10	6
Q6	0.90	0.94	0.73	0.83	23	14
Q7	0.68	0.81	0.52	0.69	31	17
Q8	0.63	0.74	0.41	0.62	54	20
Average	0.75	0.84	0.55	0.71	35	19

Table 2: Normalized precision and recall of KB vs. non-KB systems

From Table 2, it is obvious that the use of domain knowledge leads to significant improvements in both precision and recall over all the queries tested. The average improvements of KB system over non-KB system are 35% for normalized precision and 19% for normalized recall.

For completeness, Figure 7 shows the screen outputs of top 18 images retrieved using both the non-KB and the KB systems. The outputs are generated with respect to query 1 --

the Ape. From the Figure, it is clear that the KB system is able to retrieve more Ape pictures among the top ranking images.

8. Conclusions

This paper describes the design and implementation of a prototype that utilizes domain knowledge for image retrieval. Users formulate the query as a concept tree incorporating domain knowledge. A concept can be a basic concept, a higher-level concept or a concept with multiple definitions. The leaf nodes of the concept tree provide clues for their detection within the contents of the images. These clues include specifications for text, color and texture attributes. The weights computed are propagated up the concept tree to arrive at the weight of the root concept, which corresponds to the degree of similarity between the concept-based query and the image.

The prototype was tested with a set of queries chosen from a database containing over 12,000 images. To demonstrate the effectiveness of the knowledge-based system, the results are compared with a system that does not utilize domain knowledge. Experimental results show significant improvements in retrieval performance for our system over the non-knowledge-based system. The testing also demonstrates that the use of domain knowledge is a promising approach to improving the retrieval effectiveness of image retrieval systems.

Acknowledgments

The authors would like to thank the National University of Singapore for the provision of a research grant RP950658 under which this work is conducted.

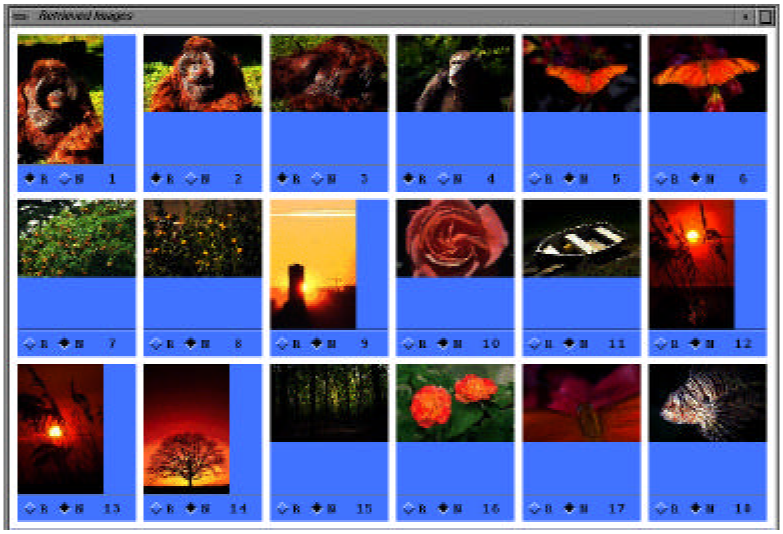
References

1. Al-Hawamdeh S., Ooi B.C., Price R., Tng T.H., Ang Y.H. and Hui L. (1991). Nearest Neighbor Searching in a Picture Archival System. Proc. of ACM Int'l Conference on Multimedia and Information System, McGraw Hill, 17-34.
2. Belkin N.J. & Croft W.B. (1987). Retrieval Techniques. Annual Review of Information Science and Technology (ARIST), 22, 1987.
3. Binaghi E., Gagliardi I. & Schettini R. (1992). Indexing and Fuzzy Logic based Retrieval of Color Images. IFIP Visual Database Systems, II, Elsevier Science Publishers B.V., Oct, 79-92.
4. Carter R.C. & Carter E.C. (1983). CIELUV Color Difference Equations for Self-luminous Displays. Color Research and Applications, 8, 252-553.

5. Chan S.K. (1994). Content-Based Image Retrieval. M.Sc Dissertation, Department of Information Systems and Computer Science, National University of Singapore.
6. Chua T.S., Lim S.K. & Pung H.K. (1994). Content-based Retrieval of Segmented Images. ACM Multimedia '94, San Francisco, Oct, 211-218.
7. Croft W.B. & Thompson R.H. (1987). IIIR: a new Approach to the Design of Document Retrieval System. Journal of ASIS, 38, 389-404.
8. Faloutsos C., Barber R., Flicker M., Hafner J., Niblack W. & Petkovic D. (1994). Efficient and Effective Querying by Image Content. Journal of Intelligent Information Systems, 3, 231-262.
9. Gudivada V.N. & Raghavan V.V. (1995). Guest Editors' Introduction: Content-based Image Retrieval Systems. IEEE Computer, 28(9), Sep, 18-22.
10. Hsu W., Chua T.S. & Pung H.K. (1995). An Integrated Color-Spatial Approach to Content-based Image Retrieval. ACM Multimedia' 95, San Francisco, Nov, 305-313.
11. Minsky M. (1967). A Framework for Representing Knowledge. In: Winston P. (Ed). The Psychology of Computer Vision, New York, McGraw Hill, 211-277.
12. Nagasaka A. & Tanaka Y. (1992). Automatic Video Indexing and Full Video Search for Object Appearances. IFIP Visual Database Systems, II, Elsevier Science Publishers B.V., Oct, 113-127.
13. Pratt W.K. (1991). Digital Image Processing (2nd Ed), John Wiley.
14. Price R., Chua T.S. & Al-Hawamdeh S. (1992). Applying Relevance Feedback to a Photo Archival System. Journal of Information Science, 18, 203-215.
15. Salton G. & McGill M.J. (1983). Introduction to Modern Information Retrieval. McGraw Hill.
16. Tamura H., Mori S. & Yamawaki T. (1978). Textural Features Corresponding to Visual Perception. IEEE Trans. Syst. Man., and Cybern, SMC-8, Jun, 460-472.
17. Teo K.C. (1995). Knowledge-based Image Retrieval. M.Sc Dissertation, Department of Information Systems & Computer Science, National University of Singapore.
18. Tong R.M., Appelbaum L.A., Askman V.N. & Cunningham J.F. (1987). Conceptual Information Retrieval Using RUBRIC. ACM Conference on Research and Development in Information Retrieval, 247-253.
19. Yoshitaka A, Kishida S, Hirakawa M & Ichikawa T (1994). Knowledge Assisted Content-Based Retrieval For Multimedia Databases. IEEE Multimedia, 12-21.

Figure 7: Top eighteen images retrieved for Query 1 (the Ape)

(7.a) The non-knowledge-based system



(7.b) The knowledge -based system