SIMPLIcity: Semantics-sensitive Integrated Matching for Picture LIbraries*

James Z. Wang**, Jia Li***, and Gio Wiederhold†

Dept. of Computer Science, Stanford University, Stanford, CA 94305, USA

Abstract. We present here SIMPLIcity (Semantics-sensitive Integrated Matching for Picture LIbraries), an image retrieval system using semantics classification and integrated region matching (IRM) based upon image segmentation. The SIMPLIcity system represents an image by a set of regions, roughly corresponding to objects, which are characterized by color, texture, shape, and location. The system classifies images into categories which are intended to distinguish semantically meaningful differences, such as textured versus nontextured, indoor versus outdoor, and graph versus photograph. Retrieval is enhanced by narrowing down the searching range in a database to a particular category and exploiting semantically-adaptive searching methods. A measure for the overall similarity between images, the IRM distance, is defined by a region-matching scheme that integrates properties of all the regions in the images. This overall similarity approach reduces the adverse effect of inaccurate segmentation, helps to clarify the semantics of a particular region, and enables a simple querying interface for region-based image retrieval systems. The application of SIMPLIcity to a database of about 200,000 general-purpose images demonstrates accurate retrieval at high speed. The system is also robust to image alterations.

1 Introduction

The need for efficient content-based image retrieval has increased tremendously in many application areas such as biomedicine, military, commerce, education, and Web image classification and searching. Content-based image retrieval is highly challenging because of the large size of the database, the difficulty of understanding images, both by people and computers, the difficulty of formulating a query, and the problem of evaluating the results. Efficient indexing and searching of large-scale image databases remains as an open problem. The automatic

* This work was supported in part by the National Science Foundation's Digital Libraries initiative. We would like to thank the help of Oscar Firschein and anonymous reviewers. An on-line demonstration is provided at `http://WWW-DB.Stanford.EDU/IMAGE/`

** Also of Biomedical Informatics, Stanford University. Email: `wangz@cs.stanford.edu`

*** Research performed when the author was with Stanford University. Currently with the Xerox Palo Alto Research Center. Email: `jiali@db.stanford.edu`

† Also of Biomedical Informatics, Stanford University. Email: `gio@cs.stanford.edu`

derivation of semantics from the content of an image is the focus of interest for research on image databases. Image *semantics* has several levels: semantic types, object composition, abstract semantics, and detailed semantics.

Many content-based image database retrieval systems have been developed, such as the IBM QBIC System [9] developed at the IBM Almaden Research Center, the Photobook System developed by the MIT Media Lab [10], the Visualseek System [13] developed at Columbia University, the WBIIS System [18] developed at Stanford University, and the Blobworld System [2] developed at U.C. Berkeley. Content-based image retrieval systems roughly fall into three categories depending on the signature extraction approach used: histogram, color layout, and region-based search. There are also systems that combine retrieval results from individual algorithms by a weighted sum matching metric [4, 9], or other merging schemes [12].

In traditional histogram-based systems [9, 11], an image is characterized by its global color histogram. The drawback of a global histogram representation is over-summarization. Information about object location, shape, and texture is discarded. Color histogram search is sensitive to intensity variation, color distortions, and cropping.

For traditional color layout indexing [9], images are partitioned into blocks and the average color or the color distribution of each block is stored. Thus, the color layout is essentially a low resolution representation of the original image. More advanced systems [18] use significant wavelet coefficients instead of averaging. By adjusting block sizes or the levels of wavelet transforms, the coarseness of a color layout representation can be tuned. The finest color layout using a single pixel block is merely the original image. We can hence view a color layout representation as an opposite extreme of a histogram, which naturally retains shape, location, and texture information if at proper resolutions. However, as with pixel representation, although information such as shape is preserved in the color layout representation, the retrieval system cannot perceive it directly. Color layout search is sensitive to shifting, cropping, scaling, and rotation because images are characterized by a set of local properties.

Region-based retrieval systems attempt to overcome the issues of color layout search by representing images at the object-level. A region-based retrieval system applies image segmentation to decompose an image into regions, which correspond to objects if the decomposition is ideal. Since the retrieval system has identified objects in the image, it is easier for the system to recognize similar objects at different locations and with different orientations and sizes. Region-based retrieval systems include the NeTra system [8], the Blobworld system [2], and the query system with color region templates [14].

The NeTra and the Blobworld systems compare images based on individual regions. Although querying based on a limited number of regions is allowed, the query is performed by merging single-region query results. Because of the great difficulty of achieving accurate segmentation, these systems tend to partition one object into several regions with none of them being representative for the object, especially for images without distinctive objects and scenes. Consequently, it is often difficult for users to determine which regions and features should be used for retrieval. Not much attention has been paid to developing similarity measures that combine information from all of the regions. One work in this direction is the querying system developed by Smith and Li [14]. The efficiency of their similarity measure depends critically on a pre-defined library of patterns, which are described only by color for the system in [14]. This measure is sensitive to object shifting, scaling, and rotation.

Although region-based systems attempt to decompose images into constituent objects, a representation composed of pictorial properties of regions is not ensured to be well related to its semantics. There is no clear mapping from a set of pictorial properties to semantics. An approximately round brown region might be a flower, an apple, a face, or a part of sunset sky. Moreover, pictorial properties such as color, shape, and texture of one object may vary dramatically in different images.

Despite the fact that it is currently impossible to reliably recognize objects in general-purpose images, there are methods to distinguish certain semantic types of images. The categorization of images into semantic types is one step towards filling the gap between pictorial representations and semantics. Information about semantic types enables a system to constrict the search range of images and improve retrieval by tuning a matching scheme to the semantic type in consideration. One example of semantics classification is the identification of natural photographs and artificial graphs generated by computer tools [5, 19]. Other examples include a system to detect objectionable images developed by Wang et al. [19], a system to classify indoor and outdoor scenes developed by Szummer and Picard [15], and a system to classify city scenes and landscape scenes [17]. Wang and Fischler [20] have shown that rough but accurate semantic understanding can be very helpful in computer vision tasks such as image stereo matching. Most of these systems use statistical classification methods based on training data.

In Section 2, the architecture of the SIMPLIcity system is presented. The region segmentation algorithm is provided in Section 3. In Section 4, the classification of images into semantics types is described. The similarity measure between images is described in Section 5. Experiments and results are provided in Section 6. We conclude in Section 7.

2 Architecture of the SIMPLIcity Retrieval System

The architecture of the SIMPLIcity system is described in Figure 1, the indexing process, and Figure 2, the querying process. During indexing, the system partitions an image into 4×4 pixel blocks and extracts a feature vector for each block. The k-means clustering algorithm is then applied to segment the image. The segmentation result is fed into a classifier that determines the semantic type of the image. An image is classified as one of the n pre-defined mutually exclusive and collectively exhaustive semantic classes. As indicated previously, examples of semantic types are indoor-outdoor, objectionable-benign, and graph-photograph

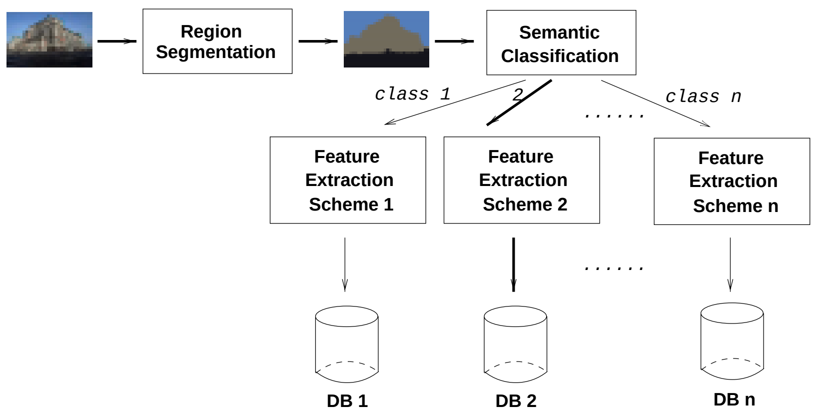


Fig. 1. The architecture of feature indexing module. The heavy lines show a sample indexing path of an image.

images. Although the classification is aimed at an image as a whole instead of individual regions, it is performed after the region segmentation so that the classifier may be assisted by information obtained from the segmentation. Features including color, texture, shape, and location information are then extracted for each region in the image. The features selected depend on the semantic type of the image. The signature of an image is the collection of features extracted from all of its regions. Signatures of images with various semantic types are stored in separate databases.

In the querying process, if the query image is not in the database, it is first passed through the same feature extraction process as was used during indexing. For an image in the database, its semantic type is first checked and then its signature is extracted from the corresponding database. Once the signature of the query image is obtained, similarity scores between the query image and images in the database with the same semantic type are computed and sorted to provide the list of images that appear to have the closest semantics.

The current implementation of the SIMPLIcity system provides several query interfaces: a CGI-based Web access interface, a JAVA-based drawing interface, a CGI-based Web interface for submitting a query image of any format anywhere on the Internet.

We allow the user to submit any images on the Internet as a query image to the system by entering the URL of an image. Our system is capable of handling any image format from anywhere on the Internet and reachable by our server via the HTTP protocol. The image is downloaded and processed by our system on-the-fly. The high efficiency of our image segmentation and matching algorithms made this feature possible[1]. To our knowledge, this feature of our system is

[1] It takes some other region-based CBIR system [2] several minutes CPU time to segment an image.

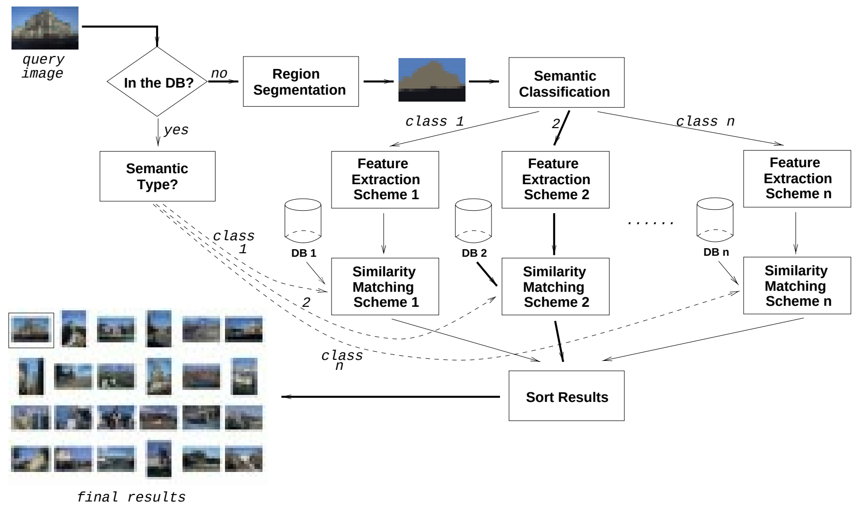


Fig. 2. The architecture of query processing module. The heavy lines show a sample querying path of an image.

unique in the sense that no other commercial or academic systems allow such queries.

3 Region Segmentation

This section describes the first component of the SIMPLIcity system: region segmentation. Our system segments images based on color and frequency features using the k-means algorithm For general-purpose images such as the images in a photo library or the images on the World-Wide Web (WWW), automatic image segmentation is almost as difficult as automatic image semantics understanding. To reduce the sensitivity to inaccurate segmentation, an integrated region matching (IRM) scheme is developed for defining a robust similarity measure.

To segment an image, SIMPLIcity partitions the image into blocks with 4×4 pixels and extracts a feature vector for each block. The k-means algorithm is applied to cluster the feature vectors into several classes each corresponding to one region in the segmented image. There are six features, three of which are the average color components in a 4×4 block. The well-known LUV color space is used, where L encodes luminance, U and V encode color information (chrominance). To obtain the other three features, a wavelet transform is applied to the L component of the image. After a one-level wavelet transform, a 4×4 block is decomposed into four frequency bands: the LL, LH, HL, and HH bands [3], each containing 2×2 coefficients. The square root of the second order moment of wavelet coefficients in each of the LH, HL, and HH bands is computed as one feature. Moments of wavelet coefficients in various frequency bands have proven effective for characterizing texture [1, 16]. The intuition behind this is that coefficients in different frequency bands show variations in different directions. For example, the HL band records activities in the horizontal direction. An image with vertical strips thus has high energy in the HL band and low energy in the LH band. This texture feature may not be the ideal feature. But it is a good compromise between computational complexity and effectiveness.

4 Image Classification

For the current implementation of the SIMPLIcity system, an image is first classified into artificial graph and photograph, which is then classified into textured and non-textured images. These three classes represent a high-level[2] categorization of images, for which the system is regarded as *semantics-sensitive*. Although we also developed a classifier to detect objectionable images, it is not integrated into the system presently because this type of images are not included in our database. By artificial graphs, we refer to synthetic images generated by computer tools, for example, clip-art images. Textured images are referred to images composed of repeated patterns that appear like a unique texture surface, e.g., a picture of lattices. Since textured images do not contain clustered objects, the perception of such images focuses on color and texture, but not shape, which is critical for understanding non-textured images. Thus an efficient retrieval system should use different features to depict those types of images. The algorithm for classifying textured and non-textured images is described in [6]. To distinguish artificial graphs and photographs, methods developed in [19, 5] are used.

5 Integrated Region Matching (IRM) Similarity Measure

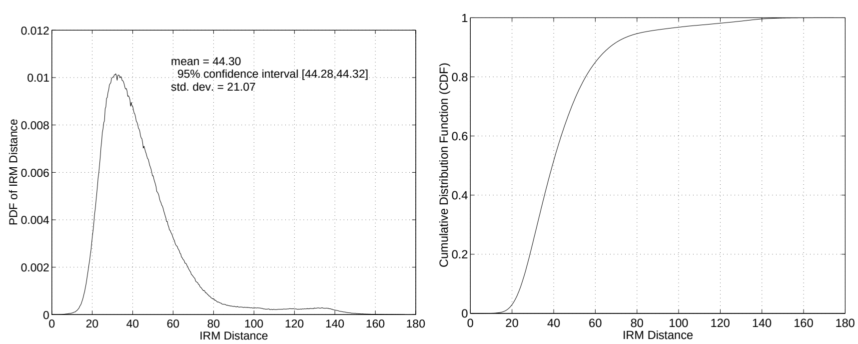
Besides using semantic classification, to reflect semantics more precisely by the region representation, the SIMPLIcity system exploits an image similarity measure determined by the properties of all the segmented regions. The motivation for fully using information about an image is that the co-existence of multiple regions often increases the confidence level of judging semantics. For example, flowers are usually present with green leaves, and boats with water. Therefore, a red region in a green background is more likely to be a flower than one in a white background. Compared with retrieval based on individual regions, the overall similarity approach reduces the influence of inaccurate segmentation. In addition to retrieval accuracy, an overall similarity measure allows a *simple* querying interface, which requires a user to specify only a query image to perform a search.

Mathematically, defining the similarity measure is equivalent to defining a distance between sets of points in a high dimensional space, i.e., the feature space. Every point in the space corresponds to the feature vector, or the descriptor, of a region. Although distance between two points in the feature space can be easily defined by the Euclidean distance, it is not obvious how to define a distance

[2] Here, we compare with low-level imagery features.

between sets of points that reflects a person's concept of semantic "closeness" of two images. A good distance is expected to take all the points in a set into account and be tolerant to inaccurate image segmentation.

To define the similarity measure, the first step is to match regions in two images. Consider the comparison of two animal photographs. The overall similarity of the two images should depend on the extent of analogousness between the animals and that between the background areas. The correspondence of objects in the images is crucial for judging similarity since it would be meaningless to compare the animal in one image with the background in another. Our matching scheme attempts to build appropriate correspondence between regions. Being aware that segmentation cannot be perfect, we "soften" the matching by allowing one region to be matched to several regions with significance scores. The principle of matching is that the closest region pair is matched first. This matching scheme is referred to as *integrated region matching* (IRM) to stress the incorporation of regions in the retrieval process. After regions are matched, the similarity measure is computed as a weighted sum of the similarity between region pairs, with weights determined by the matching scheme. Details regarding to the definitions of the IRM similarity measure and the distance between two regions are referred to [7].



Empirical PDF　　Empirical CDF

Fig. 3. The empirical PDF and CDF of the IRM distance.

To study the characteristics of the IRM distance, we performed 100 random queries on our COREL photograph data set. We obtained 5.6 million IRM distances. Based on these distances, we estimated the distribution of the IRM distance. The empirical mean of the IRM is 44.30, with a 95% confidence interval of [44.28, 44.32]. The standard deviation of the IRM is 21.07. Figure 3 shows the empirical probability distribution function and the empirical cumulative distribution function. Based on this empirical distribution of the IRM, we may give more intuitive similarity distances to the end user.

6 Experiments

The SIMPLIcity system has been implemented with a general-purpose COREL image database including about $60,000$ photographs and $140,000$ clip-art pictures, which are stored in JPEG format with size 384×256 or 256×384. These images were classified into graph, textured and non-textured types. For each image, the features, locations, and areas of all its regions were stored. Different types of images were stored in separate databases. An on-line demo is provided at URL: `http://WWW-DB.Stanford.EDU/IMAGE/SIMPLIcity/`

We compared the SIMPLIcity system with the WBIIS (Wavelet-Based Image Indexing and Searching) system [18] with the same image database. As WBIIS forms image signatures using wavelet coefficients in the lower frequency bands, it performs well with relatively smooth images, such as most landscape images. For images with details crucial to semantics, such as pictures containing people, the performance of WBIIS degrades because the multi-level wavelet transform in the system intends to smooth out details. In general, SIMPLIcity performs as well as WBIIS for smooth landscape images. For images composed of fine details, SIMPLIcity usually achieves significantly better results. For textured images, SIMPLIcity and WBIIS often perform equally well. However, in general, SIMPLIcity captures high frequency texture information better. The SIMPLIcity system also performs well on the clip-art pictures. Readers are referred to the demo web site for examples since we cannot provide many examples in this paper due to limited space.

Category	1	2	3	4	5	6	7	8	9	10
Average p	0.475	0.325	0.330	0.363	0.981	0.400	0.402	0.719	0.342	0.340
Average r	178.2	242.1	261.8	260.7	49.7	197.7	298.4	92.5	230.4	271.7
Average σ	171.9	180.0	231.4	223.4	29.2	170.7	254.9	81.5	185.8	205.8

Table 1. The average performance for each image category evaluated by precision p, the mean rank of matched images r, and the standard deviation of the ranks of matched images σ.

To provide numerical results, we tested 27 sample images chosen randomly from 9 categories, each containing 3 of the images. A retrieved image is considered a match if it belongs to the same category of the query image. Those categories are: sports and public events, beach, food, landscape with buildings, portrait, horses, tools and toys, flowers, vehicle. Most categories simply include images containing the specified objects. Images in the "sports and public events" class contain humans in a game or public event such as festival. Portraits are not included in this category. The "landscape with buildings" class refers to outdoor scenes featuring man-made constructions such as buildings and sculptures. The "beach" class refers to sceneries at coasts or river banks. For the "portrait" class, an image has to show people as the main feature. A scene with human beings as a minor part is not included.

Precisions are computed for both SIMPLIcity and WBIIS. Recalls are not calculated because the database is large and it is hard to estimate the total number of images in one category, even approximately. To account for the ranks of matched images, the average of precisions within k retrieved images, $k = 1, ..., 100$, is computed, that is, $\bar{p} = \frac{1}{100}\sum_{k=1}^{100}\frac{n_k}{k}$, where n_k is the number of matches in the first k retrieved images. This average precision is referred to as the weighted precision because it is equivalent to a weighted percentage of matched images with a larger weight assigned to an image retrieved at a higher rank. Except for the tools and toys category, in which case the two systems perform about equally well, SIMPLIcity has achieved better results measured in both ways than WBIIS. For the two categories of landscape with buildings and vehicle, the difference between the two system is quite significant—both precision and weighted precision differ by more than 0.4 on average. The average precision and weighted precision over the 27 images are 0.453 and 0.525 respectively for SIMPLIcity, but 0.226 and 0.253 for WBIIS.

The SIMPLIcity system was also evaluated based on a sub-database formed by 10 image categories, each containing 100 pictures. Within this small database, it is known whether any two images are matched. In particular, a retrieved image is considered a match if and only if it is in the same category as the query. This assumption is reasonable since the 10 categories were chosen so that each depicts a distinct semantics topic. Every image in the sub-database was tested as a query, and the retrieval ranks of all the rest images were recorded. Three statistics were computed for each query: the precision within the first 100 retrieved images, the mean rank of all the matched images, and the standard deviation of the ranks of matched images. The recall within the first 100 retrieved images was not computed because it is proportional to the precision in this special case since the total number of semantically related images for each query is fixed to be 100. The average performance for each image category in terms of the three statistics is listed in Table 1, where p denotes precision, r denotes the mean rank of matched images, and σ denotes the standard deviation of the ranks of matched images. For a system that ranks images randomly, the average p is about 0.1, and the average r is about 500.

Similar evaluation tests were carried out for color histogram match. We used LUV color space and a matching metric similar to the EMD described in [11] to extract color histogram features and match in the categorized image database. Two different color bin sizes, with an average of 13.1 and 42.6 filled color bins per image, were evaluated. We call the one with less filled color bins the Color Histogram 1 system and the other the Color Histogram 2 system. Figure 4 shows the performance as compared with the SIMPLIcity system. Both of the two color histogram-based matching systems perform much worse than the SIMPLIcity region-based CBIR system in almost all image categories. The performance of the Color Histogram 2 system is better than that of the Color Histogram 1 system due to more detailed color separation obtained with more filled bins. However, the Color Histogram 2 system is so slow that it is impossible to obtain matches on larger databases. SIMPLIcity partitions an image into an average of only 4.3

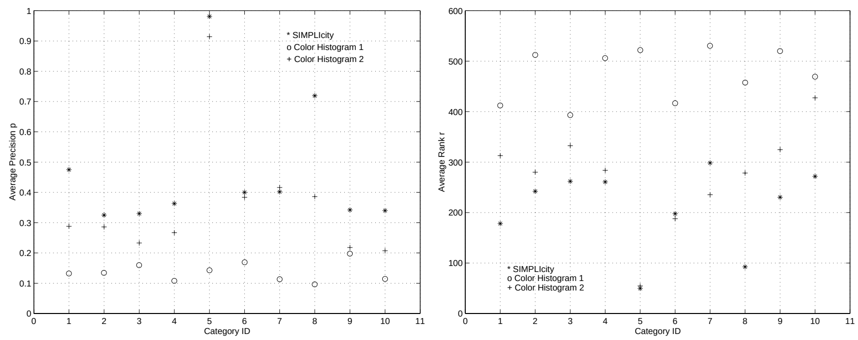


Fig. 4. Comparing with color histogram methods on average precision p and average rank of matched images r. *The lower numbers indicate better results for the second plot.*

Fig. 5. The robustness of the system to image alterations. Best 5 matches are shown. The upper-left corner is the query image. Database size: 200,000 images.

regions. It runs at about twice the speed of the faster Color Histogram 1 system and gives much better searching accuracy than the slower Color Histogram 2 system.

We have performed extensive experiments to test the robustness of the system. The system has demonstrated exceptional robustness to image alterations such as intensity variation, sharpness variation, intentional color distortions, intentional shape distortions, cropping, shifting, and rotation. Figure 5 shows some query examples, using the 200,000-image COREL database.

The algorithm has been implemented on a Pentium Pro 430MHz PC with the Linux operating system. On average, one second is needed to segment an image and to compute the features of all regions. The matching speed is very fast. When the query image is in the database, it takes about 1.5 seconds of CPU time on average to sort all the images in the database using our similarity measure. If the query is not in the database, one extra second of CPU time is spent on processing the query.

7 Conclusions and Future Work

An important contribution of this paper is the idea that images can be classified into global semantic classes, such as textured or nontextured, indoor or outdoor, objectionable or benign, graph or photograph, and that much can be gained if the feature extraction scheme is tailored to best suit each class. We have implemented this idea in SIMPLIcity (Semantics-sensitive Integrated Matching for Picture LIbraries), an image database retrieval system that uses high-level semantics classification and integrated region matching (IRM) based upon image segmentation. The application of SIMPLIcity to a database of about 200,000 general-purpose images shows fast and accurate retrieval for a large variety of images. Additionally, SIMPLIcity is robust to image alterations.

We are working on integrating more semantic classification algorithms to SIMPLIcity. In addition, it is possible to improve the accuracy by developing a more robust region-matching scheme. The speed can be improved significantly by adopting a feature clustering scheme or using a parallel query processing scheme. The system can also be extended to allow an image being classified softly into multiple classes with probability assignments. We are also working on a simple but capable interface for partial query processing. Experiments with our system on a WWW image database or a video database could be another interesting study.

We use the disk storage to store all the feature vectors in the database. On average, 400 bytes are used to store the feature vector of an image. A database of 2,000,000 images takes less than 1.0 GB of space. To further speed up the system, we may store the feature data in the main memory.

References

1. J. Bigun and J. M. H. du Buf, "N-folded symmetries by complex moments in Gabor space," IEEE-PAMI, vol. 16, no. 1, pp. 80-87, 1994.

2. C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik, "Blobworld: A system for region-based image indexing and retrieval," *Third Int. Conf. on Visual Information Systems*, June 1999.
3. I. Daubechies, *Ten Lectures on Wavelets*, Capital City Press, 1992.
4. A. Gupta and R. Jain, "Visual information retrieval," *Comm. Assoc. Comp. Mach.*, vol. 40, no. 5, pp. 70-79, May 1997.
5. J. Li, R. M. Gray, "Text and Picture Segmentation by the Distribution Analysis of Wavelet Coefficients," *Int. Conf. Image Processing*, Chicago, Oct. 1998.
6. J. Li, J. Z. Wang, G. Wiederhold, "Classification of Textured and Non-textured Images Using Region Segmentation," *Int. Conf. Image Processing*, Vancouver, Canada, Sept. 2000.
7. J. Li, J. Z. Wang, G. Wiederhold, "IRM: Integrated Region Matching for Image Retrieval," *ACM Multimedia Conf.*, Los Angeles, 2000.
8. W. Y. Ma and B. Manjunath, "NaTra: A toolbox for navigating large image databases," *Proc. IEEE Int. Conf. Image Processing*, pp. 568-71, Santa Barbara, 1997.
9. W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin, "The QBIC project: querying images by content using color, texture, and shape," *Proc. SPIE - Int. Soc. Opt. Eng., in Storage and Retrieval for Image and Video Database*, vol. 1908, pp. 173-87, 1993.
10. A. Pentland, R. W. Picard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases," *SPIE Storage and Retrieval Image and Video Databases II*, San Jose, 1995.
11. Y. Rubner, *Perceptual Metrics for Image Database Navigation*, Ph.D. Dissertation, Computer Science Department, Stanford University, May 1999.
12. G. Sheikholeslami, W. Chang, and A. Zhang, "Semantic clustering and querying on heterogeneous features for visual data," *ACM Multimedia*, pp. 3-12, Bristol, UK, 1998.
13. J. R. Smith and S.-F. Chang, "Visualseek: A fully automated content-based image query system," *Proc. Int. Conf. Image Processing*, Lausanne, Switzerland, 1996.
14. J. R. Smith and C. S. Li, "Image classification and querying using composite region templates," *Journal of Computer Vision and Image Understanding*, vol.75, no.1-2, pp. 165-74, Academic Press, 1999.
15. M. Szummer and R. W. Picard, "Indoor-outdoor image classification," *Int. Workshop on Content-based Access of Image and Video Databases*, pp. 42-51, Jan. 1998.
16. M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. Image Processing,* vol. 4, no. 11, pp. 1549-1560, Nov. 1995.
17. A. Vailaya, A. Jain, H. J. Zhang, "On image classification: city vs. landscape," *Proceedings IEEE Workshop on Content-Based Access of Image and Video Libraries*, pp. 3-8, Santa Barbara, CA, 21 June 1998.
18. J. Z. Wang, G. Wiederhold, O. Firschein, and X. W. Sha, "Content-based image indexing and searching using Daubechies' wavelets," *International Journal of Digital Libraries*, vol. 1, no. 4, pp. 311-328, 1998.
19. J. Z. Wang, J. Li, G. Wiederhold, O. Firschein, "System for screening objectionable images," *Computer Communications Journal*, vol. 21, no. 15, pp. 1355-60, Elsevier Science, 1998.
20. J. Z. Wang, M. A. Fischler, "Visual similarity, judgmental certainty and stereo correspondence," *Proceedings of DARPA Image Understanding Workshop*, Morgan Kauffman, Monterey, 1998.