

Lower-level and Higher-level Approaches to Content-based Image Retrieval *

Qasim Iqbal and J. K. Aggarwal
Computer and Vision Research Center
Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, Texas 78712, USA
aggarwaljk@mail.utexas.edu

Abstract

This paper describes a content-based image retrieval system that employs both higher-level and lower-level vision methodologies separately and in conjunction for the retrieval of images containing large manmade objects. The goal is to use the lower-level analysis module to increase the capability of the higher-level analysis module, for queries where the structure exhibited by the manmade objects is important. Higher-level analysis is performed globally to extract structure by employing the elements of perceptual grouping to extract different shape representations for higher-level feature extraction from primitive image features. The shape representations include "L" junctions, "U" junctions and parallel groups. Lower-level analysis is performed globally by using Gabor filters to extract texture features. A manmade object region of interest extracted by using perceptual grouping is used as a frame for conducting lower-level analysis. Lower-level analysis may be performed without confinement to the region of interest, i.e., over the whole image. A channel energy model is utilized to extract lower-level feature vectors consisting of fractional energies in various spatial channels. The image database consists of monocular grayscale outdoor images taken from a ground-level camera.

1 Introduction

Computer vision imposes unique requirements on the representation and manipulation of image data and knowledge [1]. Lower-level approaches analyze an image on a strictly quantitative basis for color, intensity, contrast and texture features [2]. Higher-level techniques extract semantic information that may be used by a higher-level reasoning process [1, 3]. The objective of this research is to develop and apply computer vision methods of lower-level and higher-level image analysis techniques to content-based based image retrieval (CBIR). The overall motive of this paper is to outline a framework that is capable of performing separate higher-level and lower-level queries, and in addition, a methodology for integrating higher-level and lower-level approaches in CBIR, which traditionally have largely been treated separately [1, 2, 4], for the retrieval of images containing large manmade objects (e.g. architectural objects or buildings, etc.). The goal

is to use the lower-level analysis module to increase the capability of the higher-level analysis module, for queries where the structure exhibited by the manmade objects is important.

Many research groups [5, 6, 7, 8] are actively pursuing content-based indexing, storage and retrieval of images. Some systems have already been built that provide content-based image retrieval [9, 10, 11]. These systems require user interaction, which emphasizes shape, color, and texture features to build queries. The histogram and texture analysis techniques analyze an image at a lower level on a strictly quantitative basis and are unable to capture higher-level scene descriptions that relate different primitive image features with each other. These descriptions will help in image retrieval where queries are concerned with locating images containing manmade objects. Moreover, such descriptions are relatively less sensitive to illumination changes as compared to lower-level histogram and texture analysis. Hence, we develop retrieval methodologies incorporating both lower-level and higher-level image analysis methods.

A study for the comparison of separate lower-level (histogram and texture) and higher-level (structure) approaches for the retrieval of images containing manmade objects with significant structure (represented by buildings) is presented in [2]. This paper presents an extension to that approach by integrating lower-level and higher-level approaches for the retrieval of images containing large manmade objects. However, in the current integration we do not consider the histogram. Our general framework is outlined in figure 1. An image is treated separately for two different types of analysis: global higher-level analysis and global lower-level analysis.

Higher-level analysis consists of using the general rules of *perceptual grouping*. Perceptual grouping refers to the human visual ability to extract significant image relations from lower-level primitive image features without any knowledge of the image content [1, 2, 3]. It uses such concepts as grouping by proximity, similarity, continuation, closure, and symmetry to group primitive image features into meaningful higher-level image relations. The global extraction of these higher-level features that represent significant *structure* to construct a 3-dimensional feature vector is outlined briefly in the next section. A detailed discussion of the extraction of these features may be found in [1].

Lower-level analysis is performed by employing Gabor filters to extract texture features. The texture fea-

*This research was supported in part by the Army Research Office under contracts DAAG55-98-1-0230 and DAAD19-99-1-0012 (John Hopkins University subcontract agreement 8905-48168), and by the State of Texas Higher Education Coordinating Board, Advanced Research Project 97-ARP-275.

tures are computed globally in a manmade object region of interest (ROI) extracted by using perceptual grouping. We integrate a multiscale, multiorientation *channel energy model* within the higher-level ROI, to extract a 16-dimensional feature vector consisting of fractional energies in various spatial channels. The framework is also capable of computing the texture features without confinement to the manmade ROI (i.e. the texture features may be computed for the whole image).

The underlying idea is to analyze the *structure* present in an image (using the higher-level analysis module) to make a decision regarding the presence of manmade objects. This decision is examined in an analysis block, as shown in figure 1, to use the texture represented by the structure for *refinement* and *enhancement* in the analysis of recall and precision while serving queries. Refinement refers to the process of fine tuning the set of images obtained by higher-level analysis to *eliminate* false alarms, whereas enhancement refers to the extension of the set obtained by the higher-level analysis to *include* some images that contain manmade objects that might have been missed by the higher-level analysis module. A monocular grayscale image database consisting of outdoor images taken from a ground-level camera is utilized.

The organization of the rest of the paper is as follows: section 2 details the higher-level analysis, section 3 outlines the lower-level analysis, section 4 presents the results obtained, and finally, section 5 presents the conclusions.

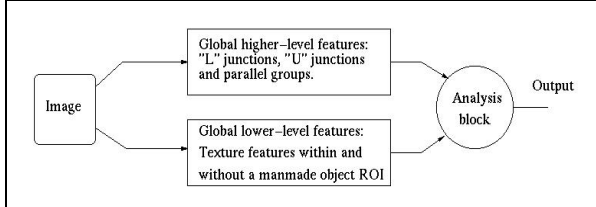


Figure 1: System overview.

2 Global higher-level feature extraction

This section describes the higher-level feature extraction process using the elements of perceptual grouping. We extract the following features hierarchically in an unconstrained environment, i.e., with no constraints on the viewing angle and depth, using the approach mentioned in [1]: *line segments, longer linear lines, “L” junctions, “U” junctions, parallel lines, parallel groups, “significant” parallel groups*. Perceptual grouping rules of similarity, continuity, and parallelism are used to extract these features.

Some of these features are self-explanatory, others are explained in the following. Longer linear lines are obtained by the extension of approximately collinear fragmented line segments that either overlap or are close to each other. The lines obtained are further pruned to eliminate lines that are very small. All other features are extracted using the longer linear lines. Parallel groups are obtained by putting constraints on the

amount of the overlaps of orthogonal projections of parallel lines and projections along the \mathcal{X} and \mathcal{Y} axes, while incorporating differences in local and intrinsic orientation of the lines. “Significant” parallel groups are extracted by further constraining the search to only those parallel groups in which at least one member line is enclosed by an “L” or “U” junction, while accommodating the obliqueness of the viewing angle.

The feature vector \mathbf{X} extracted from each image is expressed as:

$$\mathbf{X} = (\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \tilde{\mathbf{x}}_3)^t$$

where

$$\begin{aligned} \tilde{\mathbf{x}}_1 &= \frac{\text{Lines in “L” junctions}}{\text{Total \# of longer linear lines}} \\ \tilde{\mathbf{x}}_2 &= \frac{\text{Lines in “U” junctions}}{\text{Total \# of longer linear lines}} \\ \tilde{\mathbf{x}}_3 &= \frac{\text{Lines in “significant” parallel groups}}{\text{Total \# of longer linear lines}} \end{aligned} \quad (1)$$

where $\tilde{\mathbf{x}}_i \in [0, 1]$, $i \in \{1, 2, 3\}$, i.e., an image is mapped into a feature space bounded by a unit cube.

3 Global lower-level feature extraction

This section describes the procedure for lower-level feature extraction. The first stage in this process is to extract a manmade object ROI. The extraction of the ROI (which inherently is a higher-level process) is accomplished by employing perceptual grouping. The second stage globally processes only those pixels in the image that are members of the ROI, using Gabor filters to obtain lower-level texture features.

3.1 Extraction of higher-level ROI

The ROI is extracted by examining the overlapping of parallel groups and polygons. Polygons are closed figures formed by non-parallel lines. Elements of Graph Theory [12] are employed to extract polygons from an image using the cotermination graph. A set of non-parallel lines terminating at a common point is called a *cotermination*. In practice, a small neighborhood is constructed around a point to allow the lines to terminate in a small common region. The underlying idea is to take advantage of the one-to-one correspondence between the closed figures comprised of line segments and the circuits in the graph. A set of fundamental circuits is searched and extracted.

Let $G = (V, E)$ be a cotermination graph, where V and E are the set of vertices and the set of edges of G , respectively. Let $\tilde{\mathbf{e}}_{ij} \in E$ be an edge connecting vertices $\tilde{\mathbf{v}}_i, \tilde{\mathbf{v}}_j \in V$. The weight of $\tilde{\mathbf{e}}_{ij}$ is defined as $w(\tilde{\mathbf{e}}_{ij}) = \text{deg}(\tilde{\mathbf{v}}_i) + \text{deg}(\tilde{\mathbf{v}}_j)$, where $\text{deg}(\cdot)$ is the *degree* of a vertex, that is, the number of edges incident with the vertex. The edge weights are collected by extracting the adjacency matrix of the graph. The connected components of the graph are found, and each sub-graph corresponding to each component is processed separately. The weight of a spanning tree \mathcal{T} is the sum of the weights of all the branches in \mathcal{T} . We search for the maximal spanning tree, which may be found by slightly altering the minimal spanning tree algorithm to incorporate the vertices resulting in maximal-weight spanning tree [12]. The maximal spanning tree is employed

to extract the fundamental circuits. Each fundamental circuit represents a closed figure in the image, where edges on this circuit correspond to line segments on the closed figure.

A polygon is defined to be that fundamental circuit extracted that meets the following requirements [3]: (a) the polygon is simple, i.e., the edges of the polygon do not intersect among themselves, (b) the polygon is relatively compact, (c) the polygon does not have many cavities, and (d) the number of edges on the polygon does not exceed a given threshold.

3.2 Segmentation of ROI

The manmade object region of interest (ROI) is segmented by treating the parallel groups and polygons extracted as primitive structures (PS). The image is segmented to the *largest* region of manmade objects by examining the connected components of a PS graph [3]. Each node in the PS graph corresponds to one PS. Let $\mathcal{P}_i = \mathcal{C}(PS_i)$ and $\mathcal{P}_j = \mathcal{C}(PS_j)$ be convex hulls of the two PS, PS_i and PS_j , respectively, where $i \neq j$. The two nodes corresponding to PS_i and PS_j are connected in the PS graph if and only if \mathcal{P}_i and \mathcal{P}_j intersect, i.e. $\mathcal{P}_i \cap \mathcal{P}_j \neq \emptyset$. However, in case $\mathcal{P}_i \cap \mathcal{P}_j = \emptyset$, if the two convex hulls are sufficiently close, then they may be grouped together.

Let $\mathcal{P}^0 = \{\mathcal{P}_i^0 = \mathcal{C}(PS_i^0), i = 1, 2, \dots, K^0\}$ be the set of convex hulls of the PS (where the superscript denotes the current level). Let $\phi : \mathcal{P} \rightarrow \mathcal{G}$ be the function that maps the convex hull set \mathcal{P} to the PS graph \mathcal{G} . Let $\mathcal{G}^0 = (V^0, E^0) = \phi(\mathcal{P}^0)$ be the PS graph of the PS in the image, where V^0 is the set of vertices, and E^0 is the set of edges.

Let $\psi^0 = \{\psi_i^0, i = 1, 2, \dots, Z^0\}$ be the set of the connected components of \mathcal{G}^0 . Then $\psi_i^0 \subseteq \mathcal{G}^0$, and $\phi^{-1}(\psi_i^0) \subseteq \mathcal{P}^0$ is either one convex hull, or a set of convex hulls that either intersect or are close. In the former case $\phi^{-1}(\psi_i^0)$ represents one PS, PS_j^0 , hence, $PS_i^1 = PS_j^0$, and $\mathcal{P}_i^1 = \mathcal{P}_j^0$. In the latter case, the PS whose regions (convex hulls) belong to $\phi^{-1}(\psi_i^0)$ are grouped into a larger structure PS_i^1 . The region of PS_i^1 is: $\mathcal{P}_i^1 = \mathcal{C}(PS_i^1) = \mathcal{C}(\bigcup_{\mathcal{P}_j^0 \in \phi^{-1}(\psi_i^0)} \mathcal{P}_j^0)$.

A new PS graph $\mathcal{G}^1 = (V^1, E^1) = \phi(\mathcal{P}^1)$ is then established, where $\mathcal{P}^1 = \{\mathcal{P}_i^1 = \mathcal{C}(PS_i^1), i = 1, 2, \dots, K^1\}$. The connected components of \mathcal{G}^1 ($\psi^1 = \{\psi_i^1, i = 1, 2, \dots, Z^1\}$) are then found, leading to the further grouping of PS. This process continues until no PS can be further grouped. That is, the iteration stops at $\mathcal{G}^k = (V^k, E^k)$ with $E^k = \emptyset$. All potential ROIs obtained are analyzed to identify the ROI with the largest area, which is labeled as \mathcal{R} . Figure 2 displays an image and its extracted manmade object ROI.

3.3 Incorporating lower-level analysis into higher-level ROI

This section describes the integration process for incorporating lower-level analysis into the extracted higher-level ROI. After the extraction of a manmade ROI, \mathcal{R} , we proceed to limit the lower-level analysis only to \mathcal{R} . Let $\mathcal{B}(x, y)$ define the indicator function that indicates whether the pixel at location (x, y) in

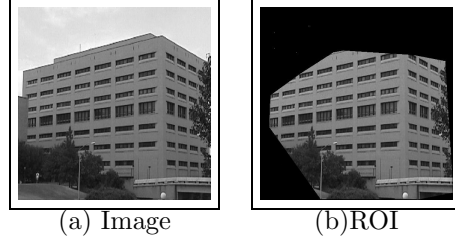


Figure 2: An image and the extracted manmade object region of interest (ROI).

the input image $I(x, y)$ is a member of \mathcal{R} , then the image $I_{\mathcal{B}}(x, y)$ comprised of only the segmented region \mathcal{R} is obtained as: $I_{\mathcal{B}}(x, y) = I(x, y) \mathcal{B}(x, y)$.

The image $I_{\mathcal{B}}(x, y)$ is treated by Gabor filters to extract spatial channel-dependent texture features. Gabor filters have been utilized for texture analysis because they have optimal joint localization (resolution) in both the spatial and the spatial frequency domains. The impulse response of an even-symmetric 2-dimensional Gabor filter is expressed as:

$$f(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2})} \cos(2\pi u_0 x) \quad (2)$$

where $f(x, y)$ represents the response at spatial locations x and y , u_0 is the frequency of a sinusoidal plane wave along the \mathcal{X} -axis (i.e., the 0° orientation), and σ_x and σ_y are the spreads of the Gaussian envelope along the \mathcal{X} and \mathcal{Y} axes, respectively. In the spatial frequency domain, the above mentioned Gabor filter is given as:

$$F(u, v) = \frac{1}{2} [e^{-\frac{1}{2}(\frac{(u-u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2})} + e^{-\frac{1}{2}(\frac{(u+u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2})}] \quad (3)$$

where $F(u, v)$ represents the response at spatial frequency locations u and v , $\sigma_u = \frac{1}{2\pi\sigma_x}$ and $\sigma_v = \frac{1}{2\pi\sigma_y}$ are the spreads of the Gaussian envelopes along the \mathcal{U} and \mathcal{V} axes, respectively.

The set of self-similar Gabor filters is obtained by appropriate rotations and scalings of $f(x, y)$ through the generating function:

$$\hat{f}_{mn}(x, y) = a^{-m} f(a^{-m}\hat{x}, a^{-m}\hat{y}), \quad a \geq 1 \quad (4)$$

where m and n are integers, $\hat{f}_{mn}(x, y)$ is the rotated and scaled version of the original filter, a is the scale factor, $n = 0, 1, \dots, N-1$ is the current orientation index, N is the total number of orientations, $m = 0, 1, \dots, M-1$ is the current scale index, M is the total number of scales, and \hat{x} and \hat{y} are the rotated coordinates:

$$\hat{x} = x \cos \theta + y \sin \theta, \quad \hat{y} = -x \sin \theta + y \cos \theta \quad (5)$$

where $\theta = \frac{n\pi}{N}$ is the orientation. The scale factor a^{-m} ensures that the filter energy is independent of m . The values of a , σ_u and σ_v are calculated as described in [2, 4].

Type	Total (T)	Retrieved (R)	Correct (C)	Recall (C/T)	Precision (C/R)
Global higher-level analysis only	45	43	36	80.00%	83.72%
Global lower-level analysis only (without ROI)	45	27	15	33.33%	55.56%
Global lower-level analysis only (with ROI)	45	46	23	51.11%	50.00%
Integrated (Refinement) without ROI	45	13	12	26.67%	92.31%
Integrated (Enhancement) without ROI	45	56	39	86.67%	69.64%
Integrated (Refinement) with ROI	45	21	18	40.00%	85.71%
Integrated (Enhancement) with ROI	45	68	41	91.11%	60.29%

Table 1: Recall and precision obtained by various methods by running a query for the retrieval of images containing buildings.

A total of 16 Gabor filters are selected, with 4 filters in different orientations at 4 different scales, i.e., $N = 4$, and $M = 4$. We select $\mathcal{U}_h = 128\sqrt{2}$ cycles per image width, and $\mathcal{U}_l = 16\sqrt{2}$ cycles per image width, resulting in $a = 2$, where \mathcal{U}_h and \mathcal{U}_l represent the upper and lower center frequencies of interest, respectively, of the Gabor filter bank. Given an image $I_{\mathcal{B}}(x, y)$, we compute

$$\hat{I}_{\mathcal{B}_{mn}}(x, y) = \mathcal{F}^{-1}[\mathcal{I}_{\mathcal{B}}(u, v) F_{mn}(u, v)] \quad (6)$$

where $\mathcal{F}^{-1}[\cdot]$ represents the inverse Fourier transform, $\hat{I}_{\mathcal{B}_{mn}}(x, y)$ represents the filtered spatial image associated with the spatial frequency channel $F_{mn}(u, v)$, $F_{mn}(u, v)$ represents the Fourier transform of $f_{mn}(x, y)$, and $\mathcal{I}_{\mathcal{B}}(u, v)$ represents the Fourier transform of the image $I_{\mathcal{B}}(x, y)$. In order to eliminate the sensitivity of the filters to absolute intensity values, we set $F_{mn}(0, 0) = 0$.

To reduce the ‘‘boundary effect’’ because of the segmentation of the ROI, a Gaussian smoothing filter is applied to all pixels inside the ROI for which the orthogonal distance between them and the nearest boundary line of the ROI is within a given threshold. Due to the localized nature of Gabor filtering, the energy leakage at the boundary of \mathcal{R} is minimized also. However, to further eliminate energy leakage outside \mathcal{R} , we set

$$\hat{I}_{\mathcal{B}\mathcal{R}_{mn}}(x, y) = \hat{I}_{\mathcal{B}_{mn}}(x, y) \mathcal{B}(x, y) \quad (7)$$

The 16-dimensional feature vector \mathbf{X} is constructed using the fractional energies in each of the 16 spatial channels, i.e.,

$$\mathbf{X} = (\tilde{\mathbf{x}}_{00}, \tilde{\mathbf{x}}_{01}, \tilde{\mathbf{x}}_{02}, \tilde{\mathbf{x}}_{03}; \tilde{\mathbf{x}}_{10}, \tilde{\mathbf{x}}_{11}, \dots, \tilde{\mathbf{x}}_{33})^t$$

where $\tilde{\mathbf{x}}_{mn}$, the fractional energy at the output of the filter in the n^{th} orientation and the m^{th} scale, is given as:

$$\tilde{\mathbf{x}}_{mn} = \frac{\sum_{y=0}^{W-1} \sum_{x=0}^{W-1} \hat{I}_{\mathcal{B}\mathcal{R}_{mn}}^2(x, y)}{\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{y=0}^{W-1} \sum_{x=0}^{W-1} \hat{I}_{\mathcal{B}\mathcal{R}_{mn}}^2(x, y)} \quad (8)$$

where W is the width of the image (the input image is square), and $\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \tilde{\mathbf{x}}_{mn} = 1$. The feature space is represented by a unit hypercube.

4 Results obtained

Experiments were conducted on an image database consisting of 150 monocular grayscale outdoor images (512×512 pixels) taken from a ground-level camera. Images in the database were assigned to three different classes based upon their structural content, viz., images containing significant structure, images containing some structure, and images containing minimal structure. In our database, we have used images of buildings to denote images containing significant structure.

The feature vectors extracted via the higher-level analysis were classified using a Bayesian classifier, whereas the feature vectors extracted using the lower-level analysis were classified using a nearest-neighbor classifier. A multivariate Gaussian class-conditional probability density function was assumed for the higher-level features, where the mean vector and the covariance matrix of the density function were obtained by using maximum likelihood estimation [1, 2]. A total of 120 images were used for testing, whereas 30 images, with 10 images in each class, were used for training.

The refinement process was accomplished by taking the logical ‘‘and’’ of the results obtained from the higher-level and lower-level analyses modules, i.e., both modules classify an image to the same class, for fine tuning. The enhancement process was accomplished by taking the logical ‘‘or’’ of the results of the two modules in the sense that at least one module classifies an image to the desired class.

Table 1 lists the results obtained for a query for the retrieval of images containing buildings. The first column shows the type of experiment performed, the second column shows the total number of images containing structures that are either fully buildings or an extended visible portion of a building (T), the third column shows the images retrieved (R), the fourth column shows the number of correct images in the set of images retrieved (C), the fifth column shows the recall (C/T), and the last column shows the precision (C/R). Recall is defined as the fraction of correct images retrieved. Precision is defined as the fraction of images retrieved that are actually correct.

The first row in table 1 shows the recall and precision obtained by only using the higher-level module. The second row shows the results obtained by using only the lower-level analysis, without the extraction of the manmade object ROI, i.e., the texture features are computed for the whole image. The third row shows the results obtained by using only the lower-level analysis, but confining it to the manmade object ROI. The

fourth and the fifth rows show the results obtained by integrating the results obtained in the first row and the second row, for refinement and enhancement, respectively, without using the ROI (whole image). The sixth and the seventh rows show the results obtained by integrating the results obtained in the first row and the third row, for refinement and enhancement, respectively, using the ROI.

The recall and precision obtained by using only the higher-level analysis module are 80.00% and 83.72%, respectively. The goal of the paper is to increase them by using a lower-level analysis module in conjunction with the higher-level analysis module. However, it must be noted that generally for a CBIR system either recall or precision may be improved on the expense of the other. As seen in the fourth and the fifth rows, without using the manmade object ROI, refinement increases the precision to 92.31% at the corresponding reduction in recall (26.67%), whereas enhancement increases the recall to 86.67% at the reduction of precision to 69.64%. The sixth and seventh row show that the use of ROI increases the precision to 85.71% at the recall of 40.00%, for refinement, and increases the recall to 91.11% at the precision of 60.29%, for enhancement.

It must be noted that results indicate that for the problem of the retrieval of images containing large manmade objects, the higher-level analysis module gives good results, and recall and precision individually may be increased by using a lower-level analysis module in conjunction with the higher-level analysis module. The justification of employing the higher-level analysis module to extract structure to serve queries regarding the retrieval of images containing manmade objects is demonstrated in figure 3. The figure shows some images from the output of a query regarding the retrieval of images containing buildings using higher-level analysis.



Figure 3: Some images from the output of a query regarding the retrieval of images containing buildings using higher-level analysis.

5 Conclusions

This paper has presented a method for the integration of global higher-level and global lower-level approaches in content-based image retrieval for the retrieval of images containing large manmade objects. The goal was to use the lower-level analysis module

to increase the capability of the higher-level analysis module for queries where the structure exhibited by the manmade objects was important. Higher-level analysis was performed by employing the general rules of perceptual grouping process to extract features that signal the presence of structure in an image. The higher-level features consisted of “L” and “U” junctions and parallel groups. The lower-level analysis was performed by first extracting a manmade object region of interest using perceptual grouping, and then treating the pixels in the region with Gabor filters for the extraction of texture features. The system provided the capability of performing the lower-level analysis without confinement to the region of interest, i.e., performing the lower-level analysis on the whole image. A channel energy model was employed to extract lower-level features that represented the fractional energies in various spatial channels. The results obtained by the higher-level analysis module, using a Bayesian classifier, were treated by the results obtained by the lower-level analysis module, using a nearest-neighbor classifier, for refinement and enhancement to the results obtained by the higher-level analysis module. The image database analyzed for serving queries consisted of monocular grayscale outdoor images taken from a ground-level camera.

References

- [1] Q. Iqbal and J. K. Aggarwal, “Applying perceptual grouping to content-based image retrieval: Building images,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 42–48, June 1999.
- [2] Q. Iqbal and J. K. Aggarwal, “Using structure in content-based image retrieval,” in *IASTED International Conference. Proceedings of the Signal and Image Processing Conference*, pp. 129–133, October 1999.
- [3] H. Q. Lu and J. K. Aggarwal, “Applying perceptual organization to the detection of man-made objects in non-urban scenes,” *Pattern Recognition*, vol. 25, no. 8, pp. 835–853, 1992.
- [4] W. Y. Ma and B. S. Manjunath, “Texture features and learning similarity,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 425–430, 1996.
- [5] R. Manmatha, “Multimedia indexing and retrieval research at the Center for Intelligent Information Retrieval,” in *Symposium on Document Image Understanding Technology*, pp. 16–30, 1997.
- [6] L. J. Guibas and C. Tomasi, “Image retrieval and robot vision research at Stanford,” in *ARPA Image Understanding Workshop*, pp. 101–108, 1996.
- [7] W. E. L. Grimson and T. Poggio, “Progress in image understanding at MIT,” in *ARPA Image Understanding Workshop*, pp. 65–74, 1996.
- [8] A. Rosenfeld, “Image understanding research at the University of Maryland,” in *ARPA Image Understanding Workshop*, pp. 57–64, 1996.
- [9] J. Ashley, R. Barber, M. Flickner, J. Hafner, D. Lee, W. Niblack, and D. Petkovic, “Automatic and semi-automatic methods for image annotation and retrieval in QBIC,” in *Proc. SPIE: Storage and Retrieval for Image and Video Databases III*, vol. 2420, pp. 24–35, Feb. 1995.
- [10] A. P. Pentland, R. Picard, and S. Sclaroff, “Photobook: Content-based manipulation of image databases,” *Int. Journal of Computer Vision*, vol. 18, no. 3, pp. 233–254, 1996.
- [11] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C.-M. Shu, “The Virage image search engine: An open framework for image management,” in *Proc. SPIE: Storage and Retrieval for Still Image and Video Databases IV*, vol. 2670, pp. 76–87, Feb. 1996.
- [12] A. Gibbons, *Algorithmic Graph Theory*. Cambridge University Press, 1985.