

IAF-00-B.6.01

OBJECT-BASED REPRESENTATIONS OF SPATIAL IMAGES

Shawn Newsam, Sitaram Bhagavathy, Charles Kenney and B.S. Manjunath
Department of Electrical and Computer Engineering
University of California, Santa Barbara

Leila Fonseca
INPE, Sao José dos Campos, Brazil

ABSTRACT

Object based representations of image data enable new content-related functionalities while facilitating management of large image databases. Developing such representations for multi-date and multi-spectral images is one of the objectives of the second phase of the Alexandria Digital Library (ADL) project at UCSB. Image segmentation and image registration are two of the main issues that are to be addressed in creating localized image representations. We present in this paper some of the recent and current work by the ADL's image processing group on robust image segmentation, registration, and the use of image texture for content representation. Built upon these technologies are techniques for managing large repositories of data. A texture thesaurus assists in creating a semantic classification of image regions. An object-based representation is proposed to facilitate data storage, retrieval, analysis, and navigation.

KEYWORDS

Spatial data, object-based representation, texture feature descriptor, image segmentation, image registration.

1. INTRODUCTION

Geo-spatially referenced images, such as satellite imagery or aerial photographs, are accumulating at an increasing rate due to recent advances in image acquisition technology. Some progress has been made in image databases in areas such as data compression and meta-data querying. However, the widening disparity between acquisition and access is preventing the full potential of the imagery from being realized. This paper discusses work that seeks to advance the current state of image database technology, particularly relating to spatial imagery.

At the heart of the problem of managing large repositories of image data is the difficulty of annotating the images for accurate and efficient access. Certain annotations of images, such as geo-referenced spatial coverage and time-stamp, can be automatically derived. However, a description of the image-content is needed before any but the most rudimentary of access methods are possible. Even if content descriptions could be unambiguously assigned by unbiased domain experts, the large manual effort required would make the undertaking futile. Annotations that can be computationally derived from the images are of much interest.

Copyright

Copyright ©2000 by the International Astronautical Federation. All rights reserved.

Much effort and some progress have been made toward deriving image content descriptors. Perhaps the most noticeable result is the upcoming MPEG-7 standardization effort whose objective is to provide a set of standardized tools to describe multimedia content [1][2][3]. However, the expectations placed on the ability of these descriptors to capture meaningful information must be tempered. The human visual system requires extensive learned experience to function as well as it does. This is especially true for an expert viewing a restricted domain of images.

This paper describes past and ongoing work relating to spatial imagery in the context of the Digital Library Projects at the University of California at Santa Barbara. We first discuss what we consider to be enabling technologies. This includes a well-established texture feature descriptor and a novel image segmentation algorithm. These technologies in turn enable higher-level functionality, such as a texture thesaurus and an object-based representation.

The rest of the paper is organized as follows. Section 2 discusses the context of this work, the Alexandria Digital Library (ADL) and Alexandria Digital Earth Prototype (ADEPT) projects. Section 3 discusses some enabling technologies. Section 4 discusses clustering and the texture thesaurus. Section 5 discusses the object-based representation and we conclude with discussions in Section 6.

2. THE UCSB DIGITAL LIBRARY PROJECTS

Most of the work presented in this paper was conducted in the context of the ADL and ADEPT projects. These projects represent phases I and II, respectively, of the US National Science Foundation funded Digital Library Initiative (DLI). ADL was one of six projects funded by phase I of the DLI which lasted from 1994 to 1998. The mission of the ADL project involves [4]:

- research on issues critical for the construction of distributed digital libraries of geo-spatially-referenced, multimedia materials;
- development of technologies necessary to support such a library;

- design, construction, and evaluation of testbed systems based on research and development results;
- resolution of organizational and technological issues underlying transition from a testbed system to an operational digital library.

One of the goals of the ADL project was to provide a working prototype of a system to access the large archive available at the Map and Imagery Lab in the UCSB library. This archive contains over five million images spanning eight decades.

The ADEPT project continues and expands the ADL work. A primary focus of the ADEPT project is to facilitate the integration of the archives and the knowledge represented therein into the teaching environment. Instructors will be able to develop *Iscares* (Information Landscapes) that will, among other things, allow students to observe the application of models to data. A simple example would be applying a rainfall model to topographic terrain data.

3. ENABLING TECHNOLOGIES

The Image Processing Group at UCSB has done a significant amount of work over the last several years developing low-level technologies for multimedia databases. A homogeneous texture descriptor has been developed and shown to be effective for a variety of image types, including satellite and aerial imagery. A novel algorithm has been developed to address the challenging problem of automatic image segmentation. Automatic methods have been developed for performing image registration.

3.1 A Homogeneous Texture Descriptor

Since an image can be considered as a mosaic of different texture regions, a homogenous texture descriptor is a likely candidate for automatic content annotation. This is especially true for satellite and aerial imagery wherein homogenous regions of terrain usually result in distinctive textures. Examples of such terrains include agricultural areas, residential areas, bodies of water, etc. The power of being able to algorithmically describe these regions based on their texture signatures is obvious. A wide range of queries become possible, such as, “Find me all regions that look like this avocado orchard.”

Our texture descriptor contains two parts. The first part relates to a perceptual characterization of texture in terms of regularity, directionality and coarseness (scale). This representation is useful for browsing applications and coarse classification of textures. It is called the Perceptual Browsing Component (PBC). The second part provides a quantitative description that can be used for accurate search and retrieval. It is called the Similarity Retrieval Component (SRC). Key features of the descriptor are:

- It captures both the high level perceptual characterization (in terms of directionality, regularity, and coarseness of a texture), as well as a robust quantitative characterization at multiple scales and orientations.
- Feature extraction is simple, involving image convolutions with a set of masks. The filters are based on a 2-D Gabor wavelet decomposition. Image convolutions can be efficiently implemented in hardware and software.

- Multiple applications can be supported by the descriptor. For example, by using PBC, browsing of image database could be performed (e.g., *show textures that are structured and are oriented at 90 deg.*). The SRC can be used for query by example type applications wherein similarity retrieval is needed.

Both components of the descriptor are derived by applying a bank of orientation- and scale-selective Gabor filters. A typical bank consists of filters tuned to four scales at six orientations (0 through 150 degrees at 30 degree intervals). The SRC is composed of the means and standard-deviations of the filter outputs so that a feature vector of length 48 results from a bank of 24 filters. A similarity measure is defined for comparing two SRC feature vectors. This similarity measure is typically taken to be the L1-norm of the difference of the vectors. A detailed description and evaluation of the SRC is given in [5] including comparisons with other texture descriptors. Specific application of the descriptor to large aerial photograph datasets is described in [6].

Figure 1 shows an example of a query-by-example of an orchard region using the SRC. The tile in the upper left is the query image. The lower ten tiles are the top retrievals.

Figure 2 shows another example of a query, this time of a highway region. Again, the top ten retrievals are shown.

The PBC captures the regularity (or lack thereof) of the texture. Its computation is based on the following observations:

- Structured textures usually consist of dominant periodic patterns.
- A periodic or repetitive pattern, if it exists, could be captured by the filtered images. This behavior is usually captured in more than one filtered output.
- The dominant scale and orientation information can also be captured by analyzing projections of the filtered images.

A detailed description of the PBC computations is available in [7].

Figure 3 shows how the PBC can be used for browsing. The textures in the top two rows are all highly structured. The textures in the bottom row of all have orientations of 30 or 120 degrees. By classifying the textures based upon perceptual characteristics, the PBC allows users to browse for images of specific regularity, orientation and scale.

A compact 12-bit version of the PBC descriptor has been accepted into the Working Draft of the MPEG-7 standard. Also accepted is a modified version of the SRC that computes the feature vector in the frequency domain.

3.2 Image Segmentation

An object-based image representation requires that the regions that correspond to objects be identified. This task is related to the well-researched and challenging problem of image segmentation. We now present a novel solution to the problem of identifying homogeneous regions in images.

3.2.1 Edge Flow[8][9]

A new boundary detection scheme based on “edge flow” is proposed to segment an image into homogeneous regions. We

assume that homogenous here refers to texture but the algorithm can use any feature descriptor, such as color, or combination of descriptors. We will also discuss segmenting large aerial images that have been partitioned into non-overlapping 64x64 pixel tiles. These images are approximately 5Kx5K pixels which results in about 6,400 tiles per image. Accurate pixel-level segmentation is often not necessary, especially for large images. The proposed scheme utilizes the Gabor texture features extracted from the image tiles and performs a coarse image segmentation based on local texture gradient. Figures 4(a)-(d) show the different stages of the segmentation algorithm, which are summarized below. Additional details can be found in [8][9].

Local texture gradient computation: Using the feature vectors, a local texture gradient is computed between each image tile and its eight surrounding neighbors. The dominant flow direction is identified in a competitive manner that is similar to a winner-takes-all representation. We call this a *texture edge flow* as the gradient information is propagated to neighboring tiles (or pixels), and this texture edge flow contains information about the (spatial) direction and energy of the local texture boundary.

Texture edge flow propagation: The local texture edge flow is propagated to its neighbors if they have the same directional preference. The flow continues until it encounters an opposite flow. This helps to localize the precise positions of the boundaries and concentrate the edge energies towards pixels where the image boundaries might exist.

Boundary detection: After the propagation reaches a stable state, the final texture edge flow energy is used for boundary detection. This is done by turning on the edge signals between two neighboring image tiles if their final texture edge flows point in opposite directions. The texture edge energy is then defined to be the summation of texture edge flow energies in the two neighboring image tiles. Figure 4(b) shows the results of this stage.

Region merging: The previous state results in many discontinuous image boundaries. They are connected to form an initial set of image regions (Figure 4(c)). At the end, a conservative region merging algorithm is used to group similar neighboring regions. The final image segmentation result is shown in Figure 4(d).

The edgflow method has been successfully used in segmenting diverse collections of images. Since this method does not require much parameter tuning, its use in automated segmentation tasks appears very promising. Further work is needed to automatically adapt to different image resolutions.

3.3 Image Registration

This section addresses the problem of registering remotely sensed images. A bottleneck in image registration has traditionally been the acquisition of control points. In remote sensing applications, users generally use manual registration which is not feasible in cases where there is a large amount of data. Thus, there is a need for automated techniques that require little or no operator supervision [10]. The most difficult registration cases are: (1) images from different sensors; (2) images taken at different times or under different conditions;

and (3) radar images. Speckle noise in radar images can produce artifacts that mimic good control points and lead to low precision or even wrong registration.

We have worked on developing automatic registration methods in order to solve the problems mentioned above. Li et al [11] have developed two algorithms for registering multi-sensor and radar images with optical images. The methods use region boundaries and other strong edges as matching primitives. Chain code correlation and other shape similarity criteria such as moments are used to match closed contours. An active contour method is used to match optical images with radar images. The methods have shown good results for registering images that contain good contours. Recently we have developed an approach for registering images taken at different times [12][13] that has been adapted for registering radar images [14]. The method essentially adopts two strategies: (1) noise removal filtering before feature extraction, and (2) incorporating multiresolution techniques into the feature extraction and matching processes. The procedure is completely automatic and relies on the gray level information content of the images and their local wavelet transform modulus maxima. The wavelet transform decomposition is used to extract feature points, which are taken as control points, and to decompose the images at different resolution levels. The correlation coefficient is used as a similarity measure and a consistency-checking step is also involved to eliminate mismatches. The algorithm is performed at progressively higher resolutions, allowing faster implementation and higher registration precision. The method has demonstrated technical feasibility for many images of forest, agriculture and urban areas from Thematic Mapper (TM), SPOT, and JERS sensors taken at different times. Figure 5 shows some registration results for TM images of the Amazon region.

4. DATA MANAGEMENT

Even though significant data compaction results from using content descriptors to reference image regions, efficient and accurate access to large image databases remains a challenge. We have investigated numerous techniques for indexing high-dimensional feature spaces. One novel approach, using clustering methods to create a texture thesaurus, is now described.

4.1 Clustering and the Texture Thesaurus

We have implemented a novel clustering-based indexing scheme for fast and accurate retrieval in large aerial image databases [6]. It is based on using Kohonen's self-organizing maps [15] to cluster the texture feature vectors so that the search space is restricted for a given query. In the resulting structure, called a *texture thesaurus*, information links are created among stored image data based on a collection of codewords and sample patterns obtained from a training set. Similar to parsing text documents using a dictionary or thesaurus, the texture information computed from images can be classified and indexed via the use of a texture thesaurus. The construction of a texture thesaurus has two stages.

The first stage uses the self-organizing maps to create clusters each of which contain visually similar patterns. This is based on a (manually) labeled set of training data. This is followed by

a hierarchical vector quantization technique to construct the texture codewords, each codeword representing a collection of texture patterns that are close to each other in the texture feature space. One can use a visual representation (image patterns whose texture descriptors are closest to the codewords) of these codewords as information samples to help users browse through the database. An iconic representation of these codewords for the aerial image database is shown in Figure 6.

The number of codewords depends on the number of distinct classes identified during the initial manual labeling and the number of texture patterns assigned to each of these classes. If a class has a large number of data points, it requires more codewords to represent all samples. This results in an unbalanced tree structure for search and indexing. An example of indexing the 2-D image features is shown in Figure 7. As can be seen, the goal of the first level of indexing tree is to identify a sub-space within which the search and retrieval should be constrained in terms of pattern similarity. On the other hand, the second level of indexing tree mainly focuses on exploring the data distribution (or density) within the sub-space, so that a set of the nearest neighbors (within the smaller cluster) can be quickly identified and retrieved.

5. OBJECT-BASED IMAGE REPRESENTATION

Manjunath and Ma [5] have used texture features extracted from images using Gabor filters for search and retrieval of image data. Deng and others [16] have come up with a dominant color scheme for content-based retrieval. Deng and Manjunath [17] have experimented with combinations of color and texture for image retrieval. Most of these methods are applied either after segmenting images into regions homogeneous in color or texture or after breaking them into smaller (usually 128x128) tiles. They do not attach importance to semantic relevance in their representation schemes. This is where the object-based scheme differs from most other image representation schemes.

In the proposed scheme, we isolate the semantically relevant portions of large images. This approach makes the query-retrieval process more convenient and has the potential to reduce the required storage space for large multi-date images..

The primary assumption is that almost all the useful information in large images is concentrated in smaller regions, which we term *objects*. Typical examples include lakes, farms, and highways in aerial pictures. Figure 8 shows two objects extracted from a large aerial image. We also assume (in accordance with common practice) that most queries will be in terms of objects. For example, common queries include “*Where can I find a lake like this?*” and “*Give me all the highways and housing colonies in this area.*”

Another assumption is that these objects are homogenous, to some extent, in color and/or texture. This assumption will help us come up with a good representation for objects.

5.1 Object Descriptors

After extracting relevant objects from aerial images, our next task is to describe these objects in such a way as to facilitate search and retrieval from a database. We use three basic

attributes for describing the objects: (1) shape, (2) color, and (3) texture.

5.1.1 Shape description

A simplistic approach to describing shape of an object is to use an *oriented-rectangle representation*. In an oriented-rectangle scheme, each object is contained within a rectangle of a certain width and height and of a certain orientation. Figure 8 shows two objects: a warehouse and a runway. The warehouse is represented by a rectangle with no tilt. The runway, on the other hand, lies within a rectangle that has a tilt with respect to the vertical axis.

A more useful shape description scheme involves the use of a *binary alpha plane*, as proposed in the MPEG-4 standard [18]. Objects are divided into rectangular macroblocks, such that they are represented with the minimum number of macroblocks within a bounding rectangle. A binary alpha plane is a binary image (of the same size as the bounding rectangle) that indicates whether or not a pixel belongs to an object. The scheme might also allow for a gray-scale alpha plane that indicates the transparency of each pixel within the object. Binary alpha planes are divided into 16 X 16 blocks. The blocks that are inside the object are signaled as opaque blocks and the blocks that are outside the object are signaled as transparent blocks. The pixels in the boundary blocks (i.e., blocks that contain pixels both inside and outside the object) are scanned in a raster scan order and coded by using context-based arithmetic coding.

Such a shape description scheme would allow us to minimize temporal redundancies in a time series of objects (say, from diurnal satellite images), by using predictive intercoding, as proposed in MPEG-4.

5.1.2 Color description

Deng and others [16] have arrived at a method of extracting the dominant colors from an image. The main methodology is to convert the color information of the image from the RGB space into the LUV space and then employ k-means clustering in the LUV space. The dominant colors are the corresponding RGB components of the statistical means of the clusters in the LUV space. The number of clusters is restricted to 15. Clusters whose statistical means are closer than a given threshold are merged so as to maintain the dominance of the clusters.

This color descriptor can be easily extended to hyperspectral images.

5.1.3 Texture description

In Section 3.1, we discussed a method for representing an image region with a single texture feature vector. Alternatively, we can combine the approaches taken in [5] and [16] to segment the texture space and, consequently, arrive at a dominant texture representation for images. We apply 24 Gabor filters of different scales and orientations on the image. We represent the filter outputs at each pixel as a 24-dimensional vector. We then cluster these vectors in the 24-dimensional space while restricting the number of clusters to 8. We also specify a merging threshold as in the dominant color extraction process. The statistical means of these clusters are the

dominant textures of the image. Figure 9(b) shows the dominant texture clusters of the warehouse in Figure 9(a).

5.2 Similarity Measure between Objects

A common type of content-based query is the *query by example*. The user provides a query image and the system retrieves all images that are “similar” to the query image.

Two objects may be similar in one or more of the following attributes: shape, color, and texture. Depending on the contextual relevance of these attributes, we can arrive at a similarity metric between two objects as a weighted combination of the distances between their attributes. The weights may be specified by the user and may depend on the source and nature of the images. For example, if the images come from both visual and infrared sensors, the user may choose to avoid comparing colors and assign a higher weight to the texture distance. If, on the other hand, the sensor suffers from severe contrast variations (which can affect texture information), the user may choose to assign a higher weight to the color distance.

5.3 The Scope for Object-based Image Representation

The object-based representation scheme has several natural advantages over conventional pixel-based schemes. The primary advantages are as follows:

Relevance of Information: In an object-based representation scheme, we consider only semantically relevant portions of images that contain most of the useful information. This makes our approach more efficient because we ignore the “irrelevant” background information.

Object-based Query and Access: Since most queries are based on objects, we can naturally expect a higher accuracy of retrieval by having an object-based representation scheme.

Storage Space Reduction: A large picture, such as a satellite image, can be represented as a mosaic of constituent objects and a background, and can thus be reconstructed given the object information. Taking advantage of the temporal redundancies that exist in multi-date objects (and images), this would lead to a considerable reduction of the space required to store an image.

Relation between Maps and Images: Images contain raw data whereas maps contain information about a geographical area. Maps portray only the important portions of a geographical area and ignore the irrelevant details. Representing aerial images in terms of the objects might enable us to find useful relationships between maps and aerial images that aid in navigation.

Spatial and Temporal Relationships: Another interesting research area is the study of spatial relationships between different objects in images. For example, we could query the database thus: “Give me all instances where a lake lies three miles east of a highway.” Another direction of research is the study of temporal relationships between objects from images taken at different points of time. A useful application of this research is the study of deforestation effects. It might also be possible to apply existing methods of object-based video coding to compress a time-series of objects. This would result in an

efficient storage scheme for satellite applications, where there are large images arriving at regular intervals of time.

6. DISCUSSIONS

An object-based decomposition and representation of images makes sense in that the useful information in most images is concentrated in smaller regions of interest. Naturally, we can expect such an approach to improve query-retrieval performance in image databases. Furthermore, we can also look forward to reducing the storage space required to store an image by storing its constituent objects instead.

7. ACKNOWLEDGEMENTS

This research was supported in part by the following grants and awards: NSF-IRI9704785 (B. S. Manjunath); NSF-IIS9817432 (S. Bhagavathy); ONR/AASERT N00014-98-1-0515 (S. Newsam); ONR N00014-96-1-0456 (C. Kenney).

8. REFERENCES

- [1] MPEG-7 Requirements Document V. 9 (F. Periera, editor), ISO/IEC JTC1/SC29/WG11/Document #N2859, July 1999, Vancouver (Canada).
- [2] MPEG-7 Context, Objective and Technical Roadmap, ISO/IEC JTC1/SC29/WG11/Document #N2861, July 1999, Vancouver (Canada).
- [3] MPEG-7 Applications Document V. 9 (A. Lindsay, editor), ISO/IEC JTC1/SC29/WG11/Document #N2860, July 1999, Vancouver (Canada).
- [4] The Alexandria Digital Library Project. <http://www.alexandria.ucsb.edu/adl.html>.
- [5] Manjunath, B.S. and Ma, W.Y., “Texture features for browsing and retrieval of image data,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(8), 837-842, 1996.
- [6] Ma, W.Y. and Manjunath, B.S., “A texture thesaurus for browsing large aerial photographs,” Journal of The American Society for Information Science, 49(7):633-648, 1998.
- [7] Manjunath, B.S.; Wu, P.; Newsam, S.; and Shin, H.D., “A Texture Descriptor for Browsing and Similarity Retrieval,” to appear in the EURASIP Image Communication Journal.
- [8] Ma, W.Y. and Manjunath, B.S., “EdgeFlow: A technique for boundary detection and segmentation,” IEEE Transactions on Image Processing (2000) (accepted for publication).
- [9] Ma, W.Y. and Manjunath, B.S., “Edge flow: a framework for boundary detection and image segmentation,” Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR’97), San Juan, Puerto Rico, pp. 744-749, June 1997.

- [10]Fonseca, L.M.G. and Manjunath, B.S, "Registration Techniques for Multisensor Remotely Sensed Imagery," Photogrammetric Engineering and Remote Sensing, v. 62, n. 9, pp. 1049-1056, Sept. 1996.
- [11]Li, H.; Manjunath, B.S.; and Mitra, S.K., "A contour-based approach to multisensor image registration." IEEE Transactions on Image Processing, v. 4, n. 3, pp. 320-334, Mar. 1995.
- [12]Fonseca, L.M.G. and Costa, M.H, "Automatic Registration of Satellite Images," Brazilian Symposium on Graphic Computation and Image Processing, 10, Campos de Jordão, 1997. Proceedings. Los Alamitos: IEEE Computer Society, 1997. pp. 219-226.
- [13]Fonseca, L.M.G.; Costa, M. H.; Manjunath, B.S.; and Kenny, C., "Automatic Registration of Satellite Imagery," Image Registration Workshop, 1, Greenbelt, MA, 1997. Proceedings. Greenbelt: NASA Goddard Space Flight Center, 1998. pp. 13-27.
- [14]Fonseca, L.M.G.; Costa, H.M.; Dutra, L.V.; and Castellari, S.P., "Automatic Registration of Radar Imagery," IEEE IGARSS, 2000 (to be published).
- [15]Kohonen, T., "The self-organizing map," Proceedings of IEEE, 78(9), pp. 1464-1480, 1990.
- [16]Deng, D.; Kenney, C.; Moore, M.S.; and Manjunath, B.S., "Peer group filtering and perceptual color image quantization," Proceedings of the IEEE International Symposium on Circuits and Systems VLSI, (ISCAS'99), Vol. 4, pp.21-4 , June 1999.
- [17]Deng, Y. and Manjunath, B.S., "Content-based search of video using color, texture and motion," Proceedings of the IEEE International Conference on Image Processing, Vol. II, pp. 534-537, Oct 1997.
- [18]Erol, B.; Dumitras, A.; and Kossentini, F., "Emerging MPEG Standards: MPEG-4 and MPEG-7," Handbook of Image and Video Processing, Ed. A. Bovik, Academic Press, Chap. 6.5, pp.615-16, 2000.

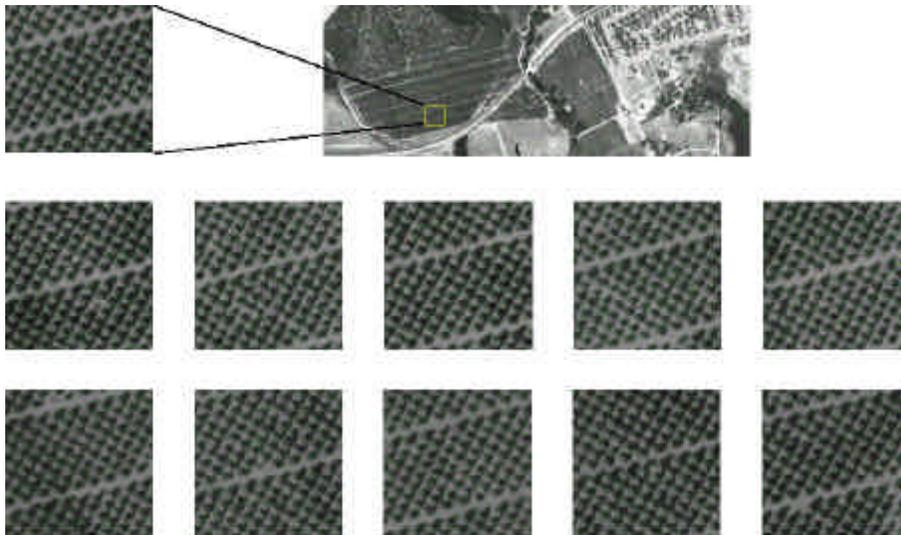


Figure 1. Query and ten retrievals using homogenous texture descriptor (SRC).

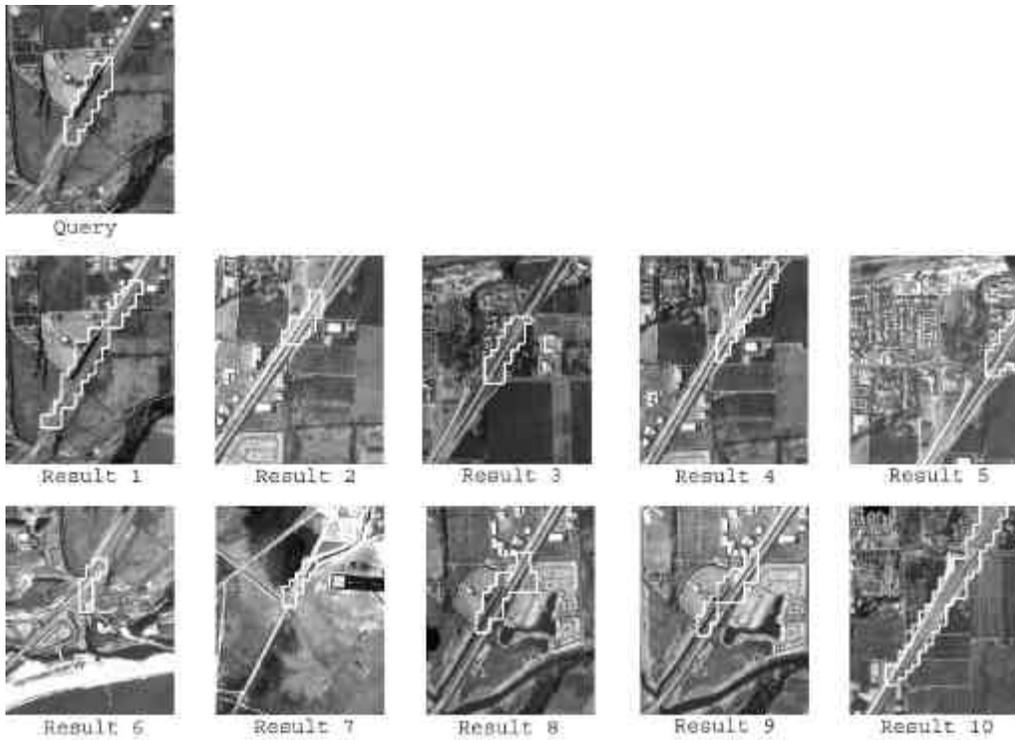


Figure 2. Query and ten retrievals using homogenous texture descriptor (SRC).

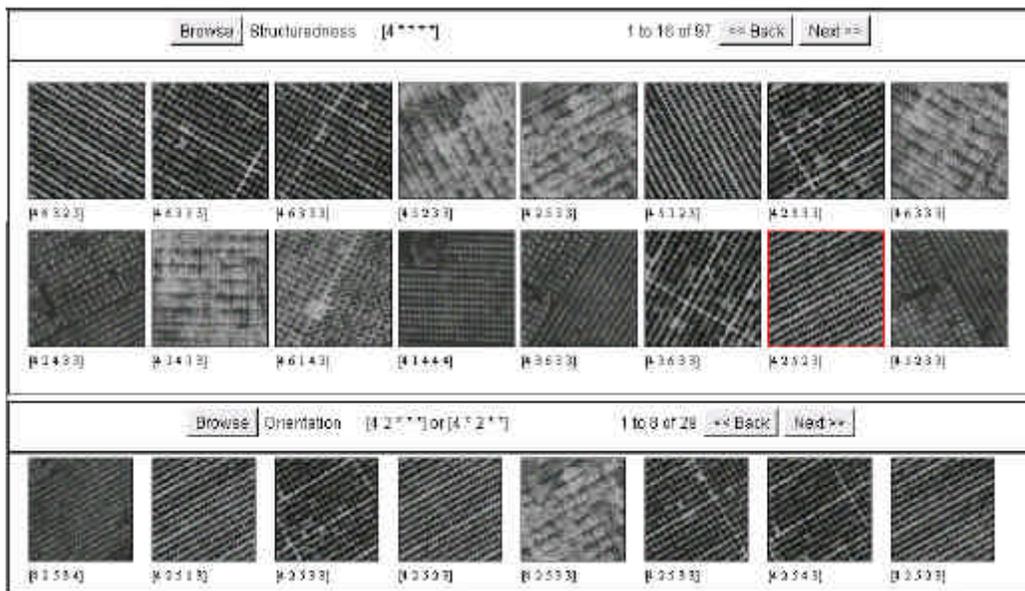


Figure 3. An example of browsing using the PBC. The top series of images are all highly structured. The bottom series of images have similar orientations.



Figure 4(a). Image segmentation based on texture flow. An aerial photograph.

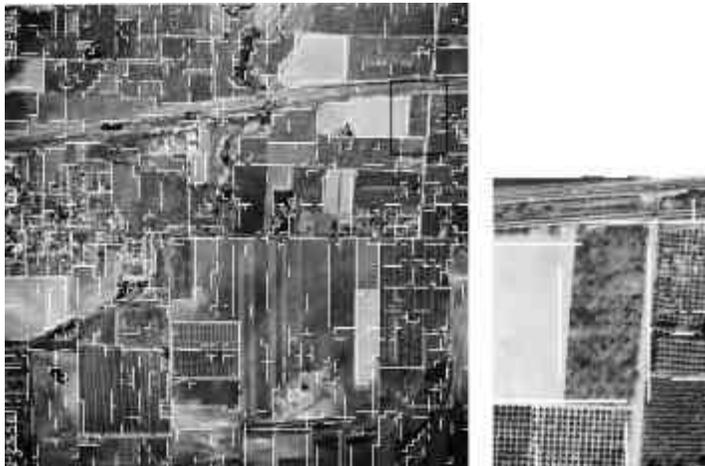


Figure 4(b). The boundaries detected by turning on the edge signals between texture flows in opposite directions.

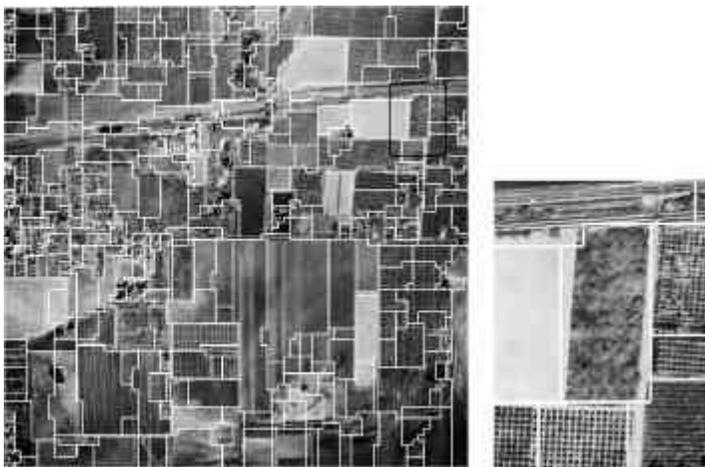


Figure 4(c). The initial set of image regions by connecting the boundaries.

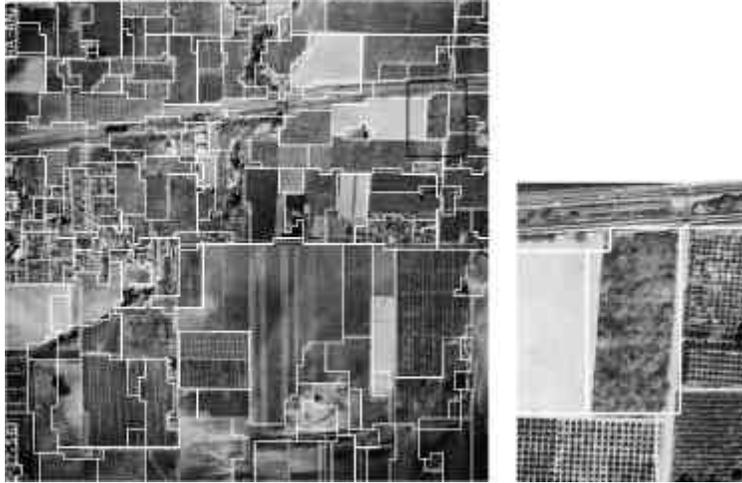


Figure 4(d). The final segmentation results after a conservative region merging.

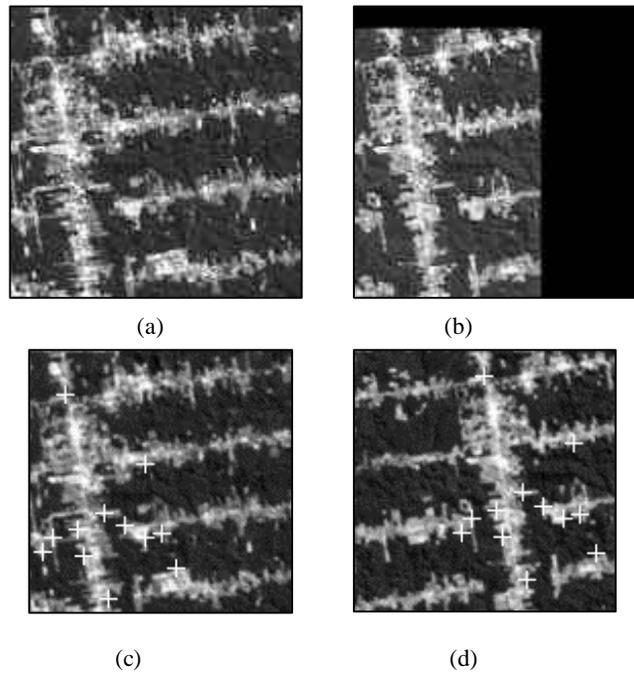


Figure 5. Amazon region images (TM): (a) reference image (07/15/94); (b) registered image (06/07/92); (c) and (d) show the initial control points superimposed on reference and warp images in the lowest level of resolution.

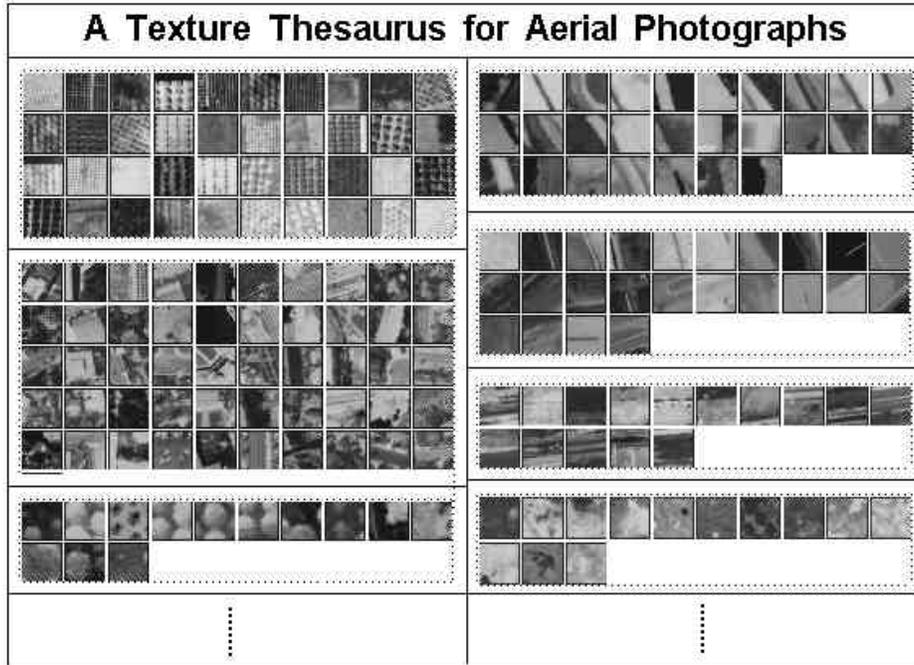


Figure 6. Examples of the codewords obtained for the aerial photographs. The patterns inside each block belong to the same class.

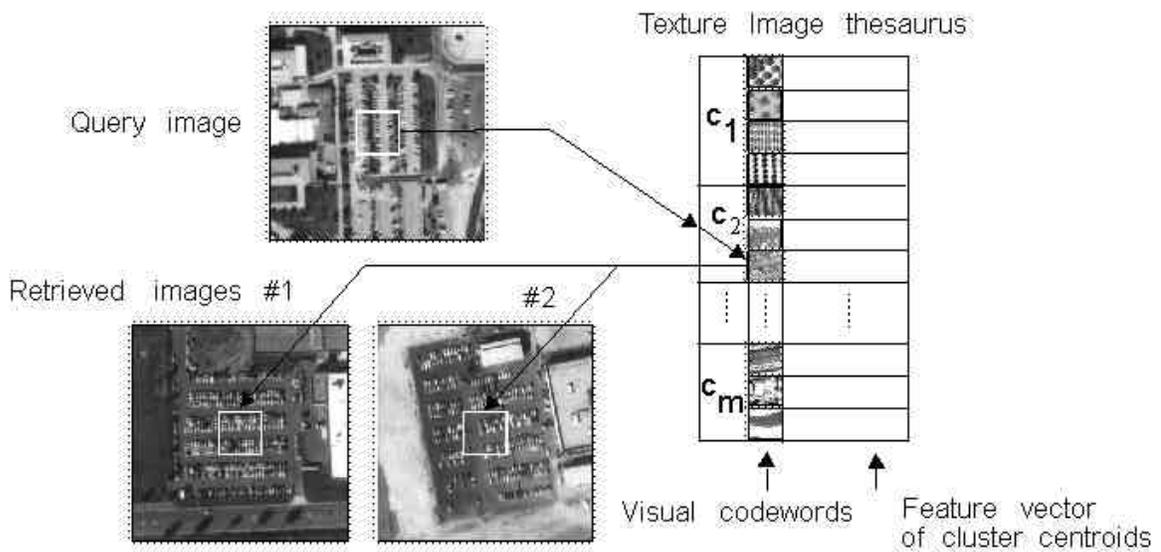


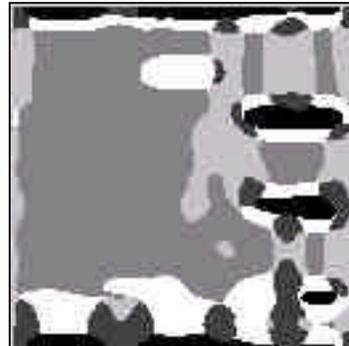
Figure 7. Using the texture thesaurus for content-based retrieval. The image tiles shown here contain parking lots.



Figure 8. Two objects (a warehouse and a runway) extracted from an aerial image.



(a)



(b)

Figure 9. The dominant texture clusters obtained from a “warehouse” image.