Modeling of Remote Sensing Image Content using Attributed Relational Graphs*

Selim Aksoy

Department of Computer Engineering, Bilkent University, Ankara, 06800, Turkey
saksoy@cs.bilkent.edu.tr

Abstract. Automatic content modeling and retrieval in remote sensing image databases are important and challenging problems. Statistical pattern recognition and computer vision algorithms concentrate on feature-based analysis and representations in pixel or region levels whereas syntactic and structural techniques focus on modeling symbolic representations for interpreting scenes. We describe a hybrid hierarchical approach for image content modeling and retrieval. First, scenes are decomposed into regions using pixel-based classifiers and an iterative split-and-merge algorithm. Next, spatial relationships of regions are computed using boundary, distance and orientation information based on different region representations. Finally, scenes are modeled using attributed relational graphs that combine region class information and spatial arrangements. We demonstrate the effectiveness of this approach in query scenarios that cannot be expressed by traditional approaches but where the proposed models can capture both feature and spatial characteristics of scenes and can retrieve similar areas according to their high-level semantic content.

1 Introduction

The constant increase in the amount of data received from satellites has made automatic content extraction and retrieval highly desired goals for effective and efficient processing of remotely sensed imagery. Most of the existing systems support building supervised or unsupervised statistical models for pixel level analysis. Even though these models improve the processing time compared to manual digitization, complete interpretation of a scene still requires a remote sensing analyst to manually interpret the pixel-based results to find high-level structures. In other words, there is still a large semantic gap between the outputs of commonly used models and high-level user expectations.

The limitations of pixel-based models and their inability in modeling spatial content motivated the research on developing algorithms for region-based analysis. Conventional region level image analysis algorithms assume that the regions

* This work was supported in part by the TUBITAK CAREER Grant 104E074 and European Commission Sixth Framework Programme Marie Curie International Reintegration Grant MIRG-CT-2005-017504. Initial work for this research was partly supported by the U.S. Army contract W9132V-04-C-0001.

consist of relatively uniform pixel feature distributions. However, complex image scenes and land structures of interest usually contain many pixels and regions that have different feature characteristics. Furthermore, two scenes with similar regions can have very different interpretations if the regions have different spatial arrangements. Even when pixels and regions can be identified correctly, manual interpretation is often necessary for many applications of remote sensing image analysis like land cover/use classification, urban mapping and monitoring, and ecological analysis in public health studies.

Symbolic representation of scenes and retrieval of images based on these representations are very challenging and popular topics in structural and syntactic pattern recognition. Previous work on symbolic representation attempted to develop languages and data structures to model the attributes and relationships of symbols/icons, and work on symbolic retrieval concentrated on finding exact or partial (inexact) matches between these representations [1, 2].

Most applications of syntactic and structural techniques to remote sensing image analysis assumed that object detection and recognition problems were solved. Using structures such as strings, graphs, semantic networks and production rules, they concentrated on the problem of interpreting the scene given the objects. Other related work in the computer vision literature used grid-based representations [3], centroids and minimum bounding rectangles [4]. Centroids and minimum bounding rectangles are useful when regions have circular or rectangular shapes but regions in natural scenes often do not follow these assumptions. Similar work can also be found in the medical imaging literature where rule-based models [5], grid-based layouts [6], and attributed relational graphs [7] were used to represent objects and their relationships given manually constructed rules or delineation of objects by experts. Most of these models are not usable due to the infeasibility of manual annotation in large volumes of images. Different structures in remote sensing images have different sizes so fixed sized grids cannot capture all structures either.

We propose a hybrid hierarchical approach for image content modeling and content-based retrieval. The analysis starts from raw data. First, pixels are labeled using Bayesian classifiers. Then, scenes are decomposed into regions using pixel-based classification results and an iterative split-and-merge algorithm. Next, resulting regions are modeled at multiple levels of complexity, and pairwise spatial relationships are computed using boundary, distance and orientation information. Finally, scenes are modeled using attributed relational graphs that combine region class information and spatial arrangements. Our work differs from other approaches in that recognition of regions and decomposition of scenes are done automatically after the system learns region models with only a small amount of supervision in terms of examples for classes of interest.

The rest of the paper is organized as follows. Decomposition of scenes into regions is described in Section 2. Modeling of regions and their spatial relationships are presented in Section 3. Scene modeling with graphs is discussed in Section 4. Using these graphs in content-based retrieval is described in Section 5 and conclusions are given in Section 6.

(a) LANDSAT scene (b) Region decomposition

Fig. 1. False color representation of a LANDSAT scene and the region decomposition obtained after applying the split-and-merge algorithm to the results of a pixel-based Bayesian classifier. White pixels in (b) represent region boundaries.

2 Scene Decomposition

The first step in scene modeling is to find meaningful and representative regions in the image. An important requirement is the delineation of each individual structure as an individual region. Automatic extraction and recognition of these regions are also required to handle large amounts of data.

In previous work [8], we used an automatic segmentation algorithm based on energy minimization, and used k-means and Gaussian mixture-based clustering algorithms to group and label the resulting regions according to their features. Our newer experiments showed that some popular density-based and graph-theoretic segmentation algorithms were not successful on our data sets because of the large amount of data and the detailed structure in multi-spectral images.

The segmentation approach we have used in this work consists of pixel-based classification and an iterative split-and-merge algorithm [9]. Bayesian classifiers that fuse information from multiple features are used to assign each pixel to one of these classes. Since pixel-based classification ignores spatial correlations, the initial segmentation may contain isolated pixels with labels different from those of their neighbors. We use an iterative algorithm that merges pixels and pixel groups using constraints on probabilities (confidence of pixel classification) and splits existing regions based on constraints on connectivity and compactness.

The algorithms proposed in this paper are evaluated using a LANDSAT scene of southern British Columbia in Canada. The false color representation of this $1,536 \times 1,536$ scene with 6 multi-spectral bands and 30 m/pixel ground resolution is shown in Fig. 1(a), and the region decomposition consisting of 1,946 regions is shown in Fig. 1(b). Spectral, textural and elevation information were used to train the Bayesian classifiers.

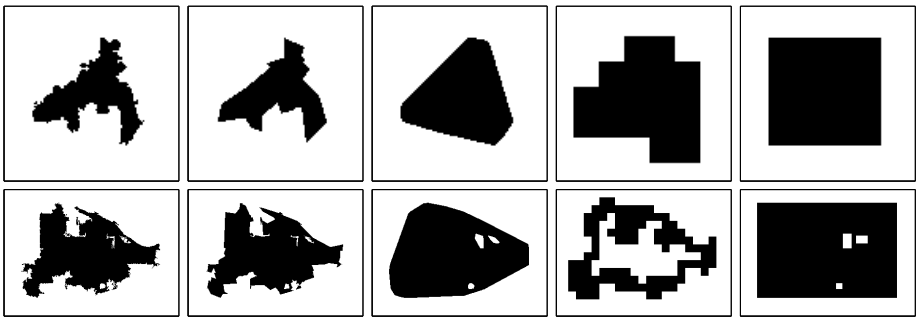


Fig. 2. Region representation examples. Rows show representations for two different regions. Columns represent, from left to right: original boundary, smoothed polygon, convex hull, grid representation, and minimum bounding rectangle.

3 Spatial Relationships

3.1 Region Modeling

A straightforward way of representing regions of an image is by using a membership array where each pixel stores the id of the region that it belongs. Hierarchical structures such as quad trees can be used to encode this membership information for faster access. Regions can also be represented using contour-based approaches such as chain codes that exploit the boundary information.

Operations on complex regions with a large number of pixels on the boundary may be computationally infeasible so regions are often modeled using approximations [10, 11]. The simplest approximation is the minimum bounding rectangle that can be useful for representing compact regions. Another simple but finer approximation is the grid representation. More detailed approximations such as polygonal representations, B-splines, or scale space representations are often necessary when operations include multiple regions.

In this work, we represent each region using its boundary chain code, polygonal representations at different smoothing levels, grid representation and minimum bounding rectangle. Regions with holes have additional lists for chain codes and polygonal approximations of their inner boundaries. Grid representation, that consists of a low-resolution grid overlaid on the region, stores all grid cells that overlap with the given region and contain at least one more region. In addition, each region has an id (unique within an image) and a label that is propagated from its pixel's class labels as described in the previous section. Example representations are given in Fig. 2. These representations at different levels of complexity are used to simplify the computation of spatial relationships between regions as described in the next section.

3.2 Pairwise Relationships

After the images are segmented and the regions are modeled at multiple levels of detail, the next step is the modeling of their spatial relationships. Regions

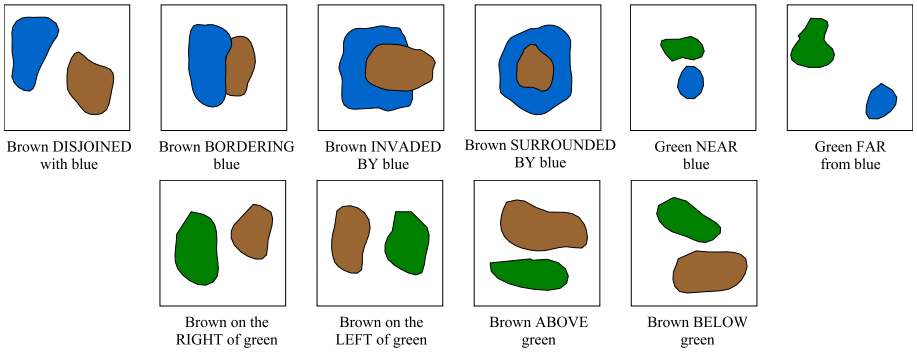


Fig. 3. Spatial relationships of region pairs: *disjoined*, *bordering*, *invaded_by*, *surrounded_by*, *near*, *far*, *right*, *left*, *above* and *below*.

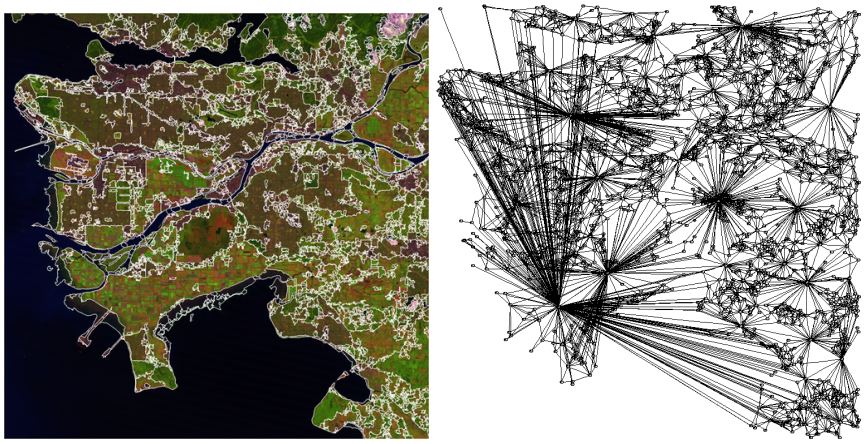
can appear in an image in many possible ways. However, regions of interest are usually the ones that are close to each other. The relationships we compute for each region pair can be grouped as boundary-class relationships (*disjoined*, *bordering*, *invaded_by*, *surrounded_by*), distance-class relationships (*near*, *far*), and orientation-class relationships (*right*, *left*, *above*, *below*) as illustrated in Fig. 3. Boundary-class relationships are based on overlaps between region boundaries. Distance-class relationships are based on distances between region boundaries. Orientation-class relationships are based on centroids of regions.

Since large scenes can easily contain thousands of regions with thousands of boundary pixels, pixel-to-pixel comparison of all possible region pairs to compute their overlaps and distances is not feasible. These computations can be significantly simplified by applying a coarse-to-fine search to find region pairs that have a potential overlap or are very close to each other. In previous work [8, 9], we used brute force comparisons of region pairs within smaller tiles obtained by dividing the original scene into manageable sized images. However, regions that occupy multiple tiles may not be handled correctly after that division. The coarse-to-fine search strategy that compares different region approximations in increasing order of complexity enables us to perform exact computations only for very close regions whereas relationships between the remaining ones are approximated using different levels of simpler boundary representations.

Since the relations between two regions can be described with multiple relationships at the same time (e.g., *invaded_by* from *left*, *bordering* from *above*, *near* and *right*), the degree of a region pair having a particular relationship is modeled using fuzzy membership functions. These relationships are based on:

- ratio of the common boundary (overlap) between two regions to the perimeter (total boundary length) of the first region,
- distance between two regions,
- angle between the horizontal (column) axis and the line joining the centroids of the regions.

Details of the membership functions are not included here due to space restrictions but more information can be found in [8].



(a) Region decomposition (b) Relationship graph

Fig. 4. Attributed relational graph of the LANDSAT scene given in Fig. 1. Region boundaries are shown here again for easy reference. Nodes are located at the centroids of the corresponding regions. Edges are drawn only for pairs that are within 10 pixels of each other to keep the graph simple.

4 Scene Modeling using Graphs

At the end of the previous section, each region pair is assigned a degree for each relationship class. In previous work [8], we modeled higher-order relationships (of region groups) by decomposing them into $\binom{k}{2}$ second-order relationships (of region pairs) combined using the fuzzy "min" operator that corresponds to the Boolean "and" operator. In this work, we model higher-order relationships using attributed relational graph (ARG) structures. ARGs are very general and powerful representations of image content. Petrakis *et al.* [7] used ARGs to represent objects and their relationships in medical images. They assumed that the regions were segmented and labeled manually, and concentrated on developing fast matching algorithms for these manually constructed graphs. However, applications of ARGs for representing contents of natural scenes have been quite limited because of inaccurate object recognition and the computational complexity of finding associations between objects in different images. Automatic decomposition of regions in Section 2 and automatic modeling of their spatial relationships in Section 3 gives us an important advantage over the existing methods that require manual segmentation and labeling of the regions.

The ARG can be adapted to model the scenes by representing regions by the graph nodes and their spatial relationships by the edges between such nodes. Nodes are labeled with the class (land cover/use) names and the corresponding confidence values (posterior probabilities) for these class assignments. Edges are labeled with the spatial relationship classes (pairwise relationship names) and the corresponding degrees (fuzzy membership values) for these relationships. The ARG for the LANDSAT scene of Fig. 1 is given in Fig. 4.

5 Scene Retrieval

When the scenes are represented using ARGs, image retrieval can be modeled as a relational matching [12] and subgraph isomorphism [13] problem. Relational matching has been extensively studied for structural pattern recognition. We use the "editing distance" [7, 14] as the (dis)similarity measure. The editing distance between two ARGs is defined as the minimum cost taken over all sequences of operations (error corrections) that transform one ARG to the other. These operations are defined as substitution, insertion and deletion. The computation of the distance between two ARGs involves not only finding a sequence of error corrections that transforms one ARG to the other, but also finding the one that yields the minimum total cost.

The retrieval scenario starts with the user's selecting of an area of interest (i.e., a set of regions) in an image. The system automatically constructs the graph for that area. Then, this graph is used to query the system to automatically find other areas (i.e., sets of regions) with similar structures in the database. In some cases, some of the relationships (e.g., *above*, *right*) can be too restrictive. Our implementation includes a relationship value named *don't_care* that allows users to constrain the searches where insertion or deletion of graph edges corresponding to relationship classes set as *don't_care* do not contribute any cost in the editing distance. Finally, resulting areas are presented to the user in increasing order of the editing distance between the subgraphs of these areas and the subgraph of the query.

Example queries are given in Figs. 5–7[1]. Traditionally, queries that consist of multiple regions are handled by averaging the features of all regions. However, this averaging causes a significant information loss because it ignores relative spatial organization and distorts the multimodal feature characteristics of the query. On the other hand, our experiments using the scene in Fig. 1 showed that the proposed ARG structure can capture both feature and spatial characteristics of region groups and can retrieve similar areas according to their high-level semantic content.

Experiments also showed that the coarse-to-fine search strategy of Section 3.2 significantly improves the performance. For the example scene with 1,946 regions shown in Fig. 1, computation of all individual region properties (boundary chain code, centroid, perimeter) took 10.56 minutes, and computation of all pairwise spatial relationships took 33.47 minutes using brute force comparisons of regions. On the other hand, computation of all additional region representations (smoothed polygon, grid representation, minimum bounding rectangle) took 2.57 seconds, and computation of all pairwise relationships took 1.7 minutes using coarse-to-fine comparisons. As for the graph search examples, the queries in Figs. 5–7 took 5.52, 7.13 and 15.96 seconds, respectively, using an unoptimized C++-based implementation on a Pentium 4, 3.0 GHz computer running Linux.

[1] Since no ground truth exists for this semantic level of analysis, we provide only qualitative examples in this paper.

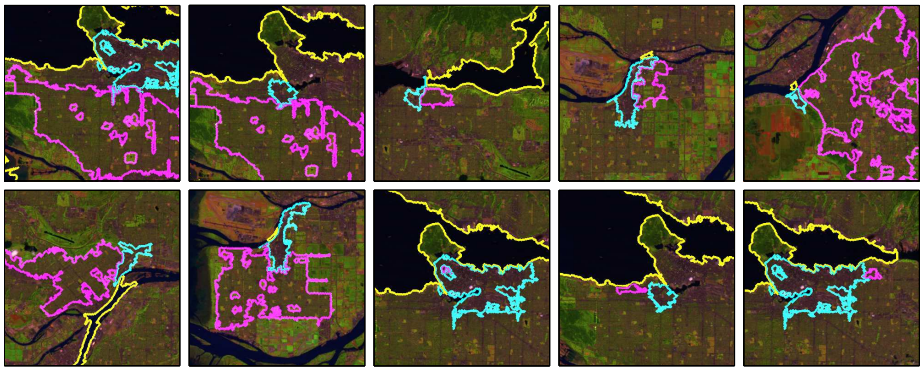


Fig. 5. Searching for a scene where a residential area is *bordering* a city center that is *bordering* water. Orientation-class is set to *don't_care*. Identified regions are marked as cyan, magenta and yellow for city, residential and water, respectively. Scenes are shown in increasing order of their editing distance to the query given on top-left.

Fig. 6. Searching for a scene where a residential area is *bordering* a field and both are *bordering* water. Identified regions are marked as cyan, magenta and yellow for residential, field and water, respectively.

6 Conclusions

We described a hybrid hierarchical approach for image content modeling that involves supervised classification of pixels, automatic grouping of pixels into contiguous regions, representing these regions at different levels of complexity, modeling their spatial relationships using fuzzy membership classes, and encoding scene content using attributed relational graph structures. We demonstrated the effectiveness of this approach for content-based retrieval using queries that provide a challenge where a mixture of spectral and textural features as well as spatial information are required for correct identification of the scenes. The

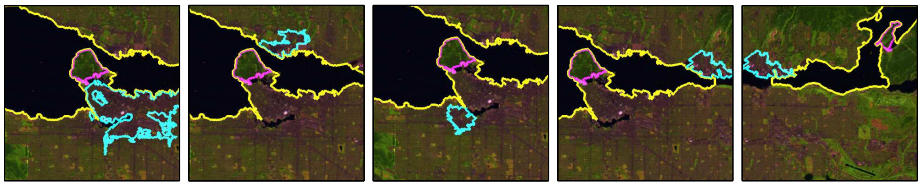


Fig. 7. Searching for a scene where a park is *invaded_by* water and a city center is *bordering* the same water. Identified regions are marked as cyan, magenta and yellow for city, park and water, respectively.

results showed that the proposed models can capture both feature and spatial characteristics of region groups and can retrieve similar areas according to their high-level semantic content. Regarding future work, we believe that improving pairwise relationship models (such as orientation-class relationships where centroids are not always very meaningful for large and non-compact regions) will make the overall representation more powerful and will prove further useful toward bridging the gap between low-level features, representations and semantic interpretation.

References

1. Dance, S., Caelli, T., Liu, Z.Q.: Picture Interpretation: A Symbolic Approach. World Scientific (1995)
2. Conte, D., Foggia, P., Sansone, C., Vento, M.: Thirty years of graph matching in pattern recognition. International Journal of Pattern Recognition and Artificial Intelligence **18** (2004) 265–298
3. Berretti, S., Bimbo, A.D., Vicario, E.: Modelling spatial relationships between colour clusters. Pattern Analysis & Applications **4** (2001) 83–92
4. Smith, J.R., Chang, S.F.: VisualSEEk: A fully automated content-based image query system. In: Proceedings of ACM International Conference on Multimedia, Boston, MA (1996) 87–98
5. Chu, W.W., Hsu, C.C., Cardenas, A.F., Taira, R.K.: Knowledge-based image retrieval with spatial and temporal constructs. IEEE Transactions on Knowledge and Data Engineering **10** (1998) 872–888
6. Tang, H.L., Hanka, R., Ip, H.H.S.: Histological image retrieval based on semantic content analysis. IEEE Transactions on Information Technology in Biomedicine **7** (2003) 26–36
7. Petrakis, E.G.M., Faloutsos, C., Lin, K.I.: Imagemap: An image indexing method based on spatial similarity. IEEE Transactions on Knowledge and Data Engineering **14** (2002) 979–987
8. Aksoy, S., Tusk, C., Koperski, K., Marchisio, G.: Scene modeling and image mining with a visual grammar. In Chen, C.H., ed.: Frontiers of Remote Sensing Information Processing. World Scientific (2003) 35–62
9. Aksoy, S., Koperski, K., Tusk, C., Marchisio, G., Tilton, J.C.: Learning Bayesian classifiers for scene classification with a visual grammar. IEEE Transactions on Geoscience and Remote Sensing **43** (2005) 581–589
10. Ballard, D.H., Brown, C.M.: Computer Vision. Prentice Hall (1982)
11. Zhang, D., Lu, G.: Review of shape representation and description techniques. Pattern Recognition **37** (2004) 1–19
12. Christmas, W.J., Kittler, J., Petrou, M.: Structural matching in computer vision using probabilistic relaxation. IEEE Transactions on Pattern Analysis and Machine Intelligence **17** (1995) 749–764
13. Messmer, B.T., Bunke, H.: Efficient subgraph isomorphism detection: A decomposition approach. IEEE Transactions on Knowledge and Data Engineering **12** (2000) 307–323
14. Myers, R., Wilson, R.C., Hancock, E.R.: Bayesian graph edit distance. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 628–635